NeOn: Lifecycle Support for Networked Ontologies

Integrated Project (IST-2005-027595)

Priority: IST-2004-2.4.7 – "Semantic-based knowledge and content systems"

D7.2.2 Revised and Enhanced Fisheries Ontologies

Deliverable Co-ordinator: **Caterina Caracciolo**

Author: **Caterina Caracciolo**

Contributor: **Aldo Gangemi**

Deliverable Co-ordinating Institution:

Food and Agriculture Organization of the United Nations (FAO)

Document Identifier:	NEON/2007/D7.2.2/v1.2	Date due:	August 31, 2007
Class Deliverable:	NEON EU-IST-2005-027595	Submission date:	August 31, 2007
Project start date:	March 1, 2006	Version:	1.2
Project duration:	4 years	State:	Final
		Distribution:	Public

NeOn Consortium

This document is part of a research project funded by the IST Programme of the Commission of the European Community, grant number IST-2005-027595. The following partners are involved in the project:

Open University (OU) – Coordinator Knowledge Media Institute – KMi Berrill Building, Walton Hall Milton Keynes, MK7 6AA United Kingdom Contact person: Martin Dzbor, Enrico Motta E-mail address: {m.dzbor, e.motta} @open.ac.uk	**Universität Karlsruhe – TH (UKARL)** Institut für Angewandte Informatik und Formale Beschreibungsverfahren – AIFB Englerstrasse 28 D-76128 Karlsruhe, Germany Contact person: Peter Haase E-mail address: pha@aifb.uni-karlsruhe.de
Universidad Politécnica de Madrid (UPM) Campus de Montegancedo 28660 Boadilla del Monte Spain Contact person: Asunción Gómez Pérez E-mail address: asun@fi.upm.es	**Software AG (SAG)** Uhlandstrasse 12 64297 Darmstadt Germany Contact person: Walter Waterfeld E-mail address: walter.waterfeld@softwareag.com
Intelligent Software Components S.A. (ISOCO) Calle de Pedro de Valdivia 10 28006 Madrid Spain Contact person: Jesús Contreras E-mail address: jcontreras@isoco.com	**Institut 'Jožef Stefan' (JSI)** Jamova 39 SI-1000 Ljubljana Slovenia Contact person: Marko Grobelnik E-mail address: marko.grobelnik@ijs.si
Institut National de Recherche en Informatique et en Automatique (INRIA) ZIRST – 655 avenue de l'Europe Montbonnot Saint Martin 38334 Saint-Ismier France Contact person: Jérôme Euzenat E-mail address: jerome.euzenat@inrialpes.fr	**University of Sheffield (USFD)** Dept. of Computer Science Regent Court 211 Portobello street S14DP Sheffield United Kingdom Contact person: Hamish Cunningham E-mail address: hamish@dcs.shef.ac.uk
Universität Koblenz-Landau (UKO-LD) Universitätsstrasse 1 56070 Koblenz Germany Contact person: Steffen Staab E-mail address: staab@uni-koblenz.de	**Consiglio Nazionale delle Ricerche (CNR)** Institute of cognitive sciences and technologies Via S. Martino della Battaglia, 44 - 00185 Roma-Lazio, Italy Contact person: Aldo Gangemi E-mail address: aldo.gangemi@istc.cnr.it
Ontoprise GmbH. (ONTO) Amalienbadstr. 36 (Raumfabrik 29) 76227 Karlsruhe Germany Contact person: Jürgen Angele E-mail address: angele@ontoprise.de	**Food and Agriculture Organization of the United Nations (FAO)** Viale delle Terme di Caracalla 1 00100 Rome Italy Contact person: Marta Iglesias E-mail address: marta.iglesias@fao.org
Atos Origin S.A. (ATOS) Calle de Albarracín, 25 28037 Madrid Spain Contact person: Tomás Pariente Lobo E-mail address: tomas.parientelobo@atosorigin.com	**Laboratorios KIN, S.A. (KIN)** C/Ciudad de Granada, 123 08018 Barcelona Spain Contact person: Antonio López E-mail address: alopez@kin.es

Work package participants

The following partners have taken an active part in the work leading to the elaboration of this document, even if they might not have directly contributed writing parts of this document:

UPM

CNR

USFD

Change Log

Version	Date	Amended by	Changes
0.1	10-07-2007	Caterina Caracciolo	First Draft
0.2	31-07-2007	Caterina Caracciolo	Completed description of domains and ontology models. Added discussions, lessons learned.
1.0	17-08-2007	Caterina Caracciolo	Revision of the entire document.
1.1	19-09-2007	Caterina Caracciolo	Implemented comments from QA
1.2	20-09-07	Andrew Bagdanov, Caterina Caracciolo	Linguistic revision. Added Annex IV provided by Aldo Gangemi, harmonized with the rest of the document

Executive Summary

This document describes and discusses the fisheries ontologies developed for use within the Fish Stock Depletion Assessment System (FSDAS). All ontologies are publicly available from the FAO website, from http://www.fao.org/aims/aos/fi. This document is organized as follows. In Chapter 1 we place our work in the context of the WP7 case study. In Chapter 2 we describe previous attempts to create ontologies for the fisheries domain. In Chapter 3 we recap the user requirements presented in our previous deliverable D7.1.1, with special attention to the functionalities needed for modelling, population and maintenance. In Chapter 4 we describe the domains and the data on which the ontologies described here are based. In Chapter 5 we describe the fisheries database where the data used to populate the fisheries ontologies is stored; we also introduce the tool used for population of the ontologies. In Chapter 6 we describe the models of all ontologies produced. In Chapter 7 we discuss some features of the ontologies. In Chapter 8 we summarize the lessons learned in the course of this work. Finally, in Chapter 9 we draw our conclusions. This document also includes four Annexes: the list of naming conventions adopted (Annex I), an essential glossary of fisheries terms (Annex II) a list of acronyms (Annex III) and a report on the conversion of the XML schema for fisheries factsheets into an ontology.

Table of Contents

List of tables

Lit of figures

1 Introduction

The WP7 case study is concerned with the creation of an ontology-driven Fisheries Stock Depletion Assessment System (FSDAS). Such a system will use FAO and non-FAO datasets on fisheries and it will access them by means of a network of ontologies.

Deliverable D7.1.1 [D7.1.1] presented the user requirements for the WP7 use case, including both the FSDAS and the lifecycle for the ontologies to be used in it. The user groups involved in the lifecycle were analyzed and special attention was paid to the requirements for ontology editors (i.e., domain experts, translators, and information management specialists) in charge of the daily maintenance of the ontologies. Visualization and editing facilities for these users were especially recommended. The next deliverable produced in WP7, Deliverable D7.2.1 [D7.2.1], was dedicated to an inventory of electronic resources available for the fisheries domain. In that deliverable, a number of resources were described, and 28 of them were selected for inclusion in the FSDAS system based on their suitability and importance. From these resources, we selected the reference data used for the collection and dissemination of fisheries statistics. Classification codes stored in the reference data are also used in the creation of XML fact sheets for fisheries. Reference data is stored in a relational database.

Relational databases are very efficient for storing and retrieving parent-child hierarchies. However, fisheries reference data is stored as a set of separate hierarchies and despite the existence of documents and knowledge evidencing relationships among elements of the various hierarchies (e.g. a biological species living in a particular sea, caught and exported in the form of different commodities). Data sitting within the various hierarchies lacks integration and relationships are neither implemented nor managed. The problem is then how to include relations across the various domains in order to allow the fisheries community around the world to better analyze the current state of fisheries and evaluate trends measuring the impact of various factors. The goal is to fully exploit the knowledge stored in the reference data to include relations across domains while keeping all functionalities currently supported.

Our approach to this issue consists in adding a semantic layer on top of the database by modelling as ontologies the domains covered by the reference data while keeping the actual data instances in the database. In this way any number of additional relationships can be handled at the ontology level, while at the same time the efficiency and all the functionalities provided by the RDBMS are kept. This solution would also allow legacy systems connected to the database to continue to work without additional effort. In order to successfully implement this solution, we need to be able to map complex database schemas to ontologies and to efficiently access the data in the database both in batch mode and at run-time. As an intermediate step, we experiment with the “static” generation of ontologies populated with data from DBs. The following two steps (run-time access to the database and the addition of relations at the ontology level) are left for future work.

Each ontology described in this deliverable is thus a “stand-alone” ontology that covers one domain (e.g., fishing areas, fisheries commodities etc.) of reference data. These ontologies are populated with data *extracted* from the database according to an ontology model created on the basis of the domain at hand and the structure of the database. This is done in order to:

1. verify that it is possible to correctly access all data from the database (i.e., no relevant piece of data in the database should remain inaccessible),
2. verify that it is possible to perform the extraction according to a sound ontological model, and

3. produce ontologies that can be used by WP7 and other NeOn partners in the next phases of the project.

However, in real life applications, it would be preferable to leave the data in its physical location (database) and access it through a semantic layer (ontologies) at run time. The reasons for preferring this setting are the following. First, since data sets can be large, it is convenient to exploit the efficiency of an RDBMS and to spare expensive and error prone processes of data conversion. Second, by keeping data in their current location, all applications that access the data will continue to work.

In deliverable D7.4.1 [D7.4.1] the three major steps in the fisheries ontologies lifecycle are described and depicted in a figure (Chapter 2) that is reproduced below (Figure 1). This figure summarizes the steps involved in the lifecycle, together with the actions that take place in each step and the user groups involved. The ontologies presented in the current deliverable did not go through the entire lifecycle, as they went from step 1 (iteration of conceptualization and population) directly to step 3 (publishing in the production environment). The intermediate step 2 (validation and update) was skipped for the twofold reason that the focus of our work was on the extraction of data instances from a relational database according to an ontological format, and because of the lack of ready-to-use tools for enabling ontology editors to validate and if necessary, to update the ontologies.

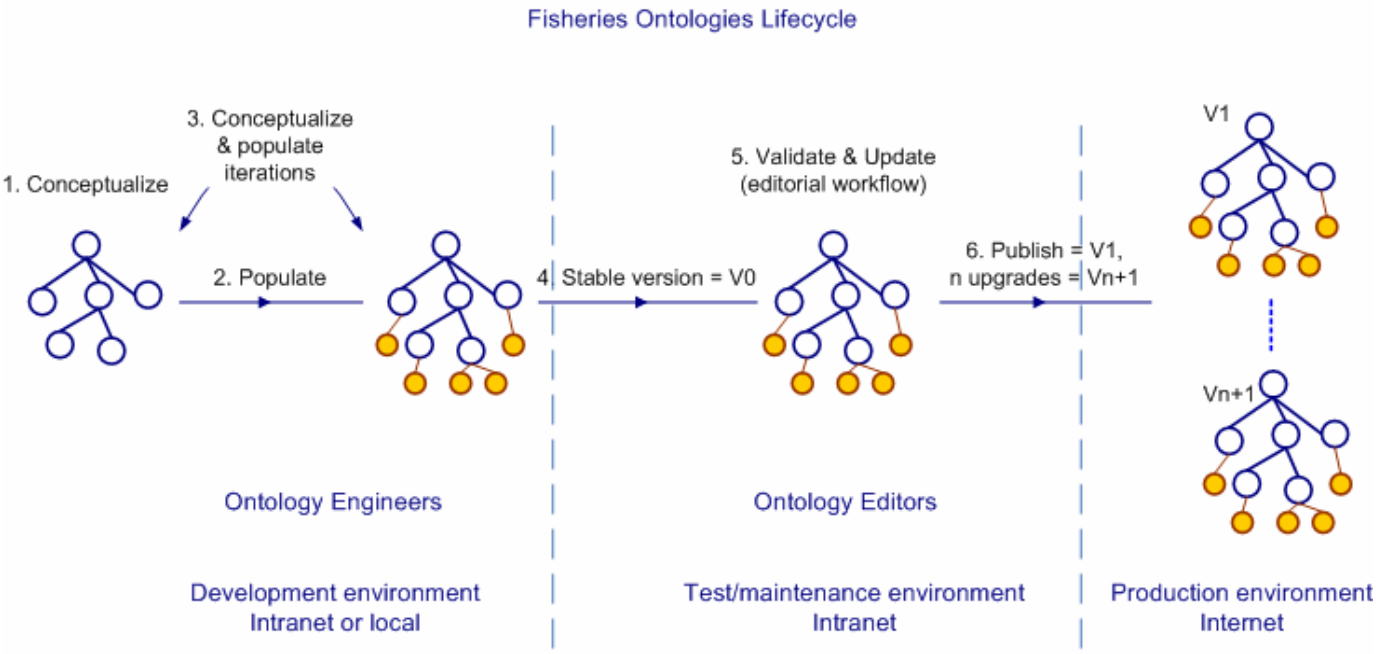


Figure 1. Major steps in the fisheries ontologies lifecycle (figure from D7.4.1, Chapter 2).

Summarizing, the ontologies presented in this deliverable constitute the first set of fisheries ontologies produced within WP7. They will be improved in a second phase of the project[1] in several ways: in terms of the way data is accessed in the database, in the amount and type of information they contain, and with links between ontologies in order to form a network.

The rest of this deliverable is organized in the following way. In Chapter 2 we survey the Fishery Ontology project (FOS), a predecessor of the NeOn project for fisheries. In Chapter 3 we provide a brief recap of requirements for tools for managing the lifecycle of ontologies in WP7. In Chapter 4 we describe the domains covered by the reference data. In Chapter 5 we describe in more detail

[1] The result of that phase will be consolidated in deliverable D7.2.3: “Enhanced networked fisheries ontologies”, due at month 30.

the structure of the database of reference data, and the tool we used to populate the ontologies from the database. The models of the ontologies created are described in detail in Chapter 6 (ontologies are published on the FAO website and so made publicly available). In Chapter 7 we highlight and discuss some relevant aspects of the ontologies produced. In Chapter 8 we summarize the lessons learned while carrying out our task, and in Chapter 9 we draw conclusions and hint at future work. The naming conventions adopted in the making of the ontologies are described in Annex I. A glossary of relevant concepts is included in Annex II, and a list of acronyms is included in Annex III. Finally, in Annex IV we report on an exploratory study on the conversion of the XML schema for fisheries factsheets into an ontology.

2 Previous work: the FOS project

The Fishery Ontology Project (FOS), in operation from 2002 to 2003, was managed jointly by ISTC-CNR and FAO [GAN04WW]. It was designed for "the creation, integration and utilization of ontologies for information integration and semantic interoperability in fisheries information systems." A detailed analysis of the FOS project can be found in deliverable D7.1.1, Annex I. Here we recap the salient aspects of that project and highlight the lessons learned from it.

At that time, integration and interoperability were interpreted as having one centralized, consistent library of ontologies that played the role of a *hub* helping the interoperability between different document servers or other information systems. The approach adopted in FOS consisted of the following steps:

1. reengineering informal or semi-structured terminological and metadata resources (KOSs: knowledge organization systems) into formal ones;
2. organizing and aligning the reengineered KOSs within an appropriate layered and modular *ontology library*.

The FOS project used the following resources (details provided refer to the time the project was carried out):

1. **OneFish topic trees [ONEF]** is a hierarchy of topics with average depth of three, organized into five disjoint categories called 'worldviews' (subjects, ecosystems, geography, species, administration), plus one worldview (stakeholder) maintained by the users of the community. Topics listed under more than one parent are marked with @.
2. **AGROVOC [AGROVOC]** is the thesaurus used in FAO to index documents related to all areas of interest of the Organization. The fragment of AGROVOC related to fisheries was used in FOS, consisting of approximately 2,000 fisheries related descriptors (out of 16,000 descriptors). The fragment was manually extracted;
3. **ASFA thesaurus [ASFA]** is the thesaurus used to index the Aquatic Science and Fisheries Abstracts (ASFA) collection of documents, which covers the world's literature on the science, technology, management, and conservation of marine, brackish water, and freshwater resources and environments, including their socio-economic and legal aspects.
4. **FIGIS reference tables [RT]** is the dataset used by FAO to reference statistical data on fisheries. At the time of the project, it included approximately 200 top-level concepts, with maximal depth of 4. It also contains 30,000 'objects' (mixed concepts and individuals), relations (specialized for each top category, but scarcely instantiated) and multilingual support.

The approach followed in FOS embodied a three-layered structure: a *foundational layer*, a *core layer* and a *domain layer*. The idea of this three-layered structure is that each top-class or property in domain ontology is a subclass of a class resp. property in the core ontology, and each top-class or property in the core ontology is a subclass of a class resp. property in the foundational ontology. For example, Yellow Tuna (domain ontology) rdfs:subClassOf Biological entity (core ontology), which rdfs:subClassOf Physical entity (foundational ontology).

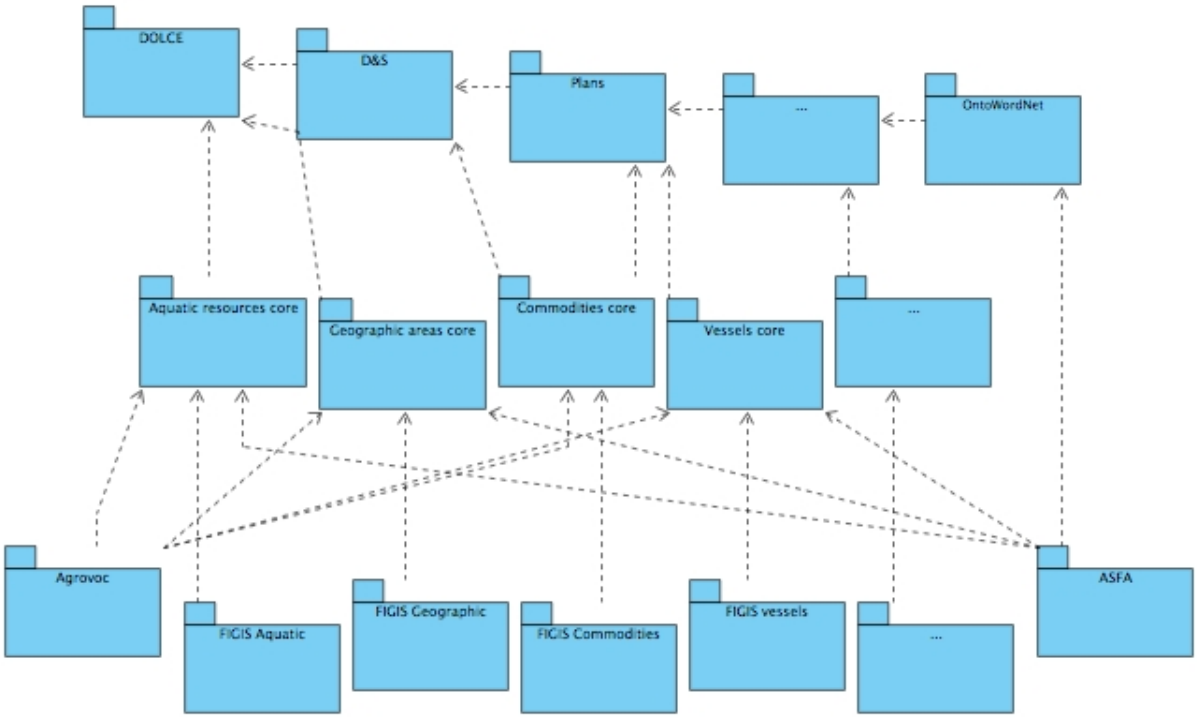


Figure 2. A diagram representing the three-layered structure of FOS.

Figure I depicts the dependencies between some of the ontologies from the three layers of the FOS library. The topmost layer contains the reused *foundational* ontologies. The middle layer shows the *core* ontologies of fisheries, the bottom layer shows some *domain* fisheries ontologies that contain dependencies to the core ones.

After the completion of FOS, the resources on which the domain ontologies were built continued to evolve, while the fisheries ontologies remained static. This happened because FOS ontologies did not enter the editorial cycle used by the people who maintained the original resources and were barely integrated into the network of resources of the organization. Moreover, since there was no mechanism to automatically reflect updates from the sources to the ontologies (and back), it was natural for editors to keep working on the sources and avoid duplicating addition/change of material. Summarizing, we identified the following limitations in FOS:

1. lack of an automatic way to maintain the mapping between FOS ontologies and the underlying resources when updated. In fact, the mappings were stored in conversion tables without an implemented mechanism to maintain the mappings over the dynamics of both ontologies and sources;

2. lack of integrated tools to create the domain ontologies by starting from the original resources. A formal workflow was defined, but not implemented in a set of integrated, smoothly-working tools;

3. lack of automatic methods to extract “modules” from the original resources and create the ontologies.

The lessons learned from the FOS project where then integrated in the user requirements for the ontology lifecycle gathered in deliverable D7.1.1. We summarize the relevant requirements for the present deliverable in the next section.

3 Requirements on modelling, population and maintenance

In this section we recap the requirements introduced in Deliverable D7.1.1 [D7.1.1] concerning the tools needed for the lifecycle of fisheries ontologies. We distinguish two groups of requirements: those concerning modelling and population of ontologies and those concerning their maintenance. The former group of activities is typically performed by ontology engineers (cf. [D7.1.1] Section 4.3), the latter by ontology editors (cf. [D7.1.1] Section 4.4).

3.1 Modelling and population

The entire process of ontology modelling should be supported. This includes functionalities for:

1. visualizing ontology elements (i.e., classes, datatype and object properties, URIs and metadata) of one or more ontologies at a time. The visualized ontologies may or may not be connected in a network;
2. visualizing mappings between ontologies and between ontologies and underlying resources such as databases;
3. editing all ontology elements of one or more ontologies at a time;
4. automatically creating and updating documentation concerning the main features of the ontology (e.g., names of all ontology elements including mappings, metadata and summarizing statistics);
5. linking to relational databases in order to allow the creation and population of ontologies based on data residing in them.

We stress the fact that an appropriate visualization is important both for population by manual editing and for population by connection to databases.

3.2 Maintenance

Tools for ontology maintenance are crucial to keeping the ontologies up-to-date and so ensuring that they can be used in real applications. It is therefore important that:

1. in case of manual maintenance (editing), all ontology elements can be manually edited and visualized, and that the appropriate metadata (e.g., author and date of the change) is stored together with its history;
2. in case of automatic population from a database, the database should be automatically checked for updates and a list of changes presented to the editor;
3. ontology versioning be supported;
4. mapping between ontologies be supported and all functionalities of editing, visualization and versioning available to it;
5. ontologies can be part of a workflow. In particular, more than one editor should be allowed to work (i.e. edit and visualize) on the same ontology, if possible with constraints (permissions) imposed by user profiles and associated with entire ontologies and modules in it.

4 Reference data

In the inventory presented in deliverable D7.2.1, 28 systems including both FAO and non FAO resources, were carefully detailed. Based on that analysis, we selected the FAO resources at the core of many information systems in fisheries: the reference data used to collect, store and access statistical data and to produce XML fact sheets on fisheries.

The FAO Fisheries and Aquaculture Information and Statistics Service (FIES) collates statistics concerning several aspects of fisheries. A time series is a sequence of observations which are ordered in time and/or space. FIES collects observations about captures, aquaculture production, catches, fleets, trade of commodities, and consumption [FISTAT]. Each piece of statistical data is referenced by the following dimensions: time (in years), space (land and/or water areas), and the variable representing the observed object (e.g., biological species). In the case of statistics concerning trade, also the "trade flow" (import/export) is included. Table 1 provides an example of statistics relative to import and export of "tunas, skipjack and Atlantic bonito, prepared or preserved" in the year 2000.

Land Area	Trade flow	Commodity	2000
Algeria	Export	Tunas, skipjack and Atlantic bonito, prepared or preserved	1
	Import	Tunas, skipjack and Atlantic bonito, prepared or preserved	841
Total Algeria			842
Grand total			842

Table 1. Import and export of fisheries commodities in Algeria in the year 2000.

The data used to indicate the dimensions are called reference data and are organized into Reference Tables (RT) [RT]. Reference tables store the *codes* assigned to reference data according to one or more coding system maintained by international organizations. They also store the association between codes and names in one or more languages (usually English, French and Spanish). Correspondence between languages is 1-1 because it results from international agreements (e.g. on names of territories, on commodities classification). Detailed information regarding fisheries statistics can be found in the Handbook of Fishery Statistical Standards [HBFSS] by the Coordinating Working Party on Fishery Statistics (CWP).[2] The entire system that manages the RT is called Rerefence Tables Management System (RTMS), whose core is an Oracle database, called FIGIS.

Reference data is also used in the fisheries fact sheets [FS] where a large amount of information about fisheries, aquaculture and related subjects, including fishing techniques, fishing areas, fisheries and aquaculture country profiles, is made available to the public in the form of semi-structured text. All fisheries fact sheets in FAO are in XML format, structured according to a comprehensive XML schema [FSschema] that includes all elements used in all types of fact sheets. Fact sheets are organized by domains (e.g., Cultured species, Fishing equipment, Fishery, Gear type), each corresponding to an element under the root FIGISdoc, the root of any fact sheet (XML document). Domains are fully specified by means of nested elements. Each element includes a description meant for human use.

[2] The Coordinating Working Party on Fishery Statistics (CWP) supported by its participating organizations has served since 1960 as the premier international and inter-organization forum for agreeing upon common definitions, classifications and standards for the collection of fishery statistics.

The schema makes use of existing standard element sets such as Dublin Core [DC], Extended Dublin Core [EDC], AGMES [AGMES] and AIDA [AIDA]. It also incorporates wherever possible existing classification schemes (such as ISO standards for countries, currencies, languages, and other fisheries-related international classification schemes) most of which are stored in the RT.

It is important to note that the schema was conceived as a means for editors to create structured documentation, and as such was not created based on a relational or ontologicial model, but was rather organised following hierarchical document formatting conventions. A dictionary of the elements used in the schema is available online [FSdic].

In the rest of this chapter we describe in detail each hierarchy of reference data used to generate the ontologies described in Chapter 6.

4.1 Land areas

Most fisheries statistics are on production and catch and are reported by individual countries. Data can then be aggregated above the national level into groups defined according to different criteria, such as geographic or economic unit. Continents, such as Africa and Asia, are typical geographical regions; the Caribbean Community (CARICOM), the Union Economique et Monetaire Ouest Africaine (UEMOA), and the Gulf Cooperation Council (GCC) are examples of economic regions.[3]

Codes used for land areas are the ISO-3166 ALPHA-2 [ISO2] and ALPHA-3 [ISO3] codes maintained by the International Standard Organization (ISO), and the M49 [M49] code maintained by the UN Statistical Division.

The names of territories (countries and groups) are established by international agreements. By agreement, two types of names of territory are given in each language: long names to be used in official documents, and short names to be used in informal communications.

Since territories and groups change over time, the RT also includes their range of validity in order to continue to be able to query the statistical database according to territories no longer existing. For example, territories can join together, as in the case of East Germany and West Germany, both created in 1949 and dissolved in 1990 to become Germany, or split as in the case of Serbia and Montenegro that in 2006 split into Serbia *and* Montenegro. Groups of territories are also dynamic, as geographical groups (continents) “change” when their member territories change, and economical groups similarly change every time a country joins or leaves a group.

4.2 Fishing areas

Marine and inland waters are divided into regions, or "FAO division areas," for the purpose of data collection and statistical reporting (for the entire list of FAO divisions, see [FAOdiv]). The FAO division areas consist of major areas, divided into sub-areas, each divided into divisions, and these finally into sub-divisions. This division of water areas forms a strict and complete hierarchy based on inclusion, or part-of. Water areas have names in natural language only at the area level, while internal divisions are given numeric names.

The FAO code used for these areas is a taxonomic code.[4] For example, the major area “SouthEast Pacific” has code 87, and one of its subdivisions has code: 87.2.1.1. Figure 3 represents the FAO major Fishing area 51, Western Indian Ocean, and its subareas, numbered from 1 to 8.

[3] For a list of regional economic organizations with which FAO works, the reader can refer to [FAO-groups].

[4] Any library user is familiar with taxonomic codes because of the Dewey classification system. In that system the digits composing the number do not signify numbers, but should be interpreted according to the classification system.

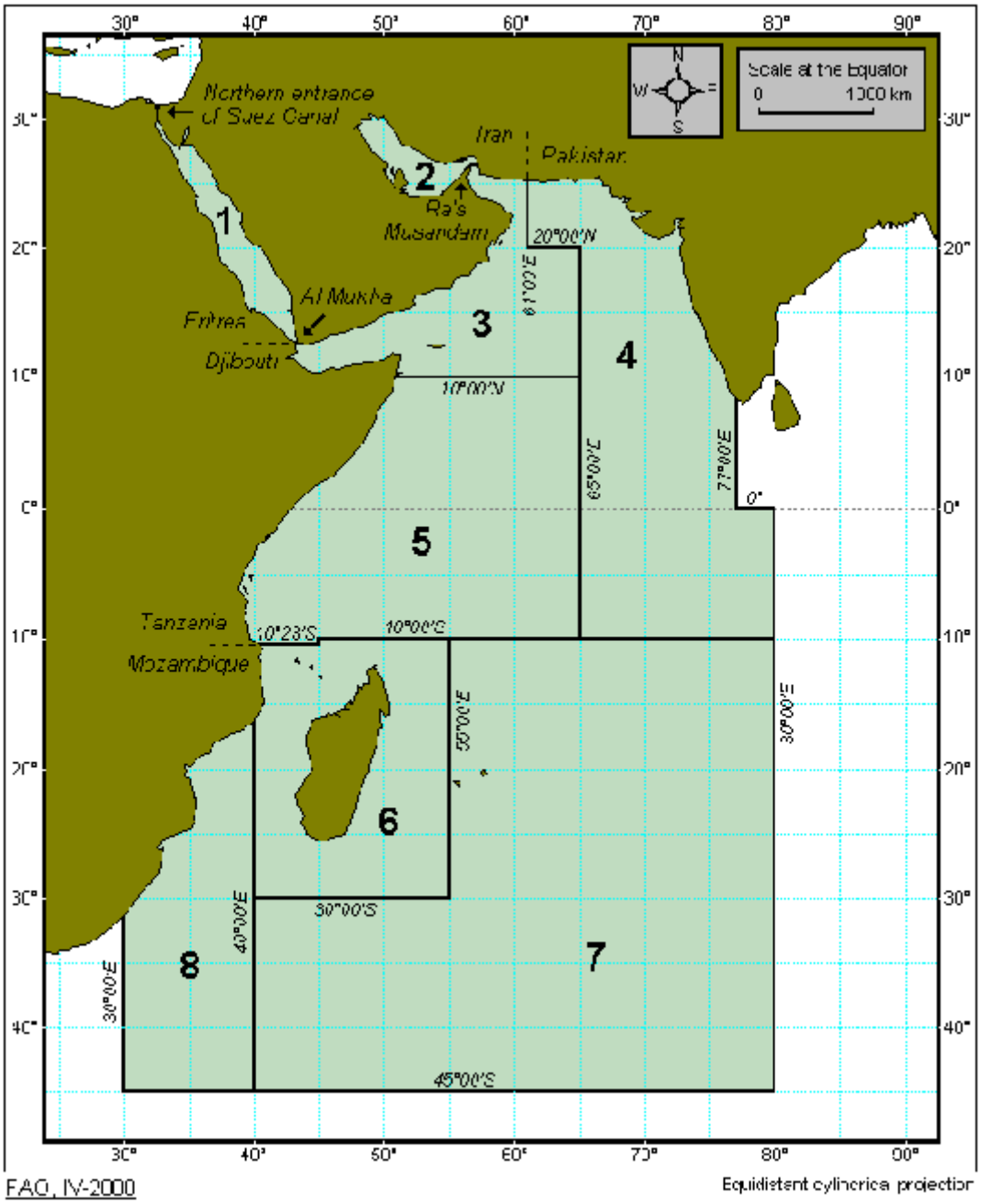


Figure 3. An example of FAO major fishing area: Western Indian Ocean (FAO code 51).

4.3 Biological entities

Reference data about biological species is used for a number of statistics, such as catch, production and trade. These statistics are collated at either the species or higher taxonomic levels. Each level is referred to as a *species items*. Species items are organized and maintained in the Aquatic Science and Fisheries Information System (ASFIS). For each species item a taxonomic code and the ISO ALPHA-3 [ISO3] are provided. An English name is available for most of the

records, and about one third of them have also a French and Spanish name.[5] Currently, the ASFIS list includes nearly 11.000 species items selected according to their interest or relation to fisheries and aquaculture.

A taxonomic code for biological entities is a 10-digit code that for any entity specifies its *type,* i.e., if it is a major group, an order, a family, a genus or a species, and its complete hierarchical path. Table 2 analyzes an example of taxonomic code (of a biological species: 1750400301) to show how the 10 digits of the taxonomic code are organized (the family "above" that species has taxonomic code: 17504XXXXX).

	Main grouping	*Order or high taxonomic level*	*Family*	*Genus*	*Species*
Digits	digit 1	digits 2 and 3	digits 4 and 5	digits 6, 7, 8	digits 9, 10
Example	1	75	04	003	01

Table 2. Structure of the 10-digit taxonomic code used for biological entities.

Since a biological entity (i.e., main group, order, family, genus, species) is only included in the reference table if/when there is data associated with it, the taxonomic classification included in the database may not be complete (i.e., there can be species for which only the main group they belong to is specified, and no order nor family are given).

The 3-alpha code identifier is a unique code made of three letters that is widely used for the exchange of data with national correspondents and among fisheries agencies.

4.4 Fisheries commodities

Fisheries commodities cover products derived from any aquatic animal (fish, crustaceans, molluscs) and residues caught for commercial, industrial or subsistence uses, by all types of classes of fishing units operating in inland, fresh and brackish waters, in inshore, offshore or high seas fishing areas.

Several coding and classification systems are available for fisheries commodities. FAO's International Standard Statistical Classification of Fishery Commodities (ISSCFC) [ISSCFC] is used for detailed information on countries or zones. The ISSCFC is an expansion of the United Nations Standard International Trade Classification (SITC) [SITC3], developed by the United Nations' Statistical Office on the basis of earlier international work on the subject. The ISSCFC is linked with the Harmonized Commodity Description and Coding System (abbreviated to HS) [HS07] of the World Customs Organization (WCO).[6] The ISSCAAP classification is also used (see section above).

The ISSCFC is a taxonomic classification system maintained by FAO and used to collect data on commodities from countries. Its maximum depth is six levels. The Harmonized System (HS) is

[5] Member agencies of the CWP have agreed to use these standard species names in statistical publications and questionnaires. However, (a) it has not been possible to assign appropriate names in all three languages to all species items, and (b) these names may not correspond with nationally or regionally-used common names.

[6] The system was originally developed by the Customs Cooperation Council (CCC), now known as the World Customs Organization (WCO). The WCO, located in Brussels, is an international organization consisting of representatives of about 139 countries and territories.

intended to serve as a universally accepted classification system for goods so countries can administer customs programs and collect trade data on exports and imports. It was designed to replace the varied tracking methods used by countries and create one common classification system with which to track trade and apply tariffs. The basic system is a 3-level taxonomic code forming a 6-digit number identifying basic commodities. Each country is allowed to add additional digits for statistical purposes (called HS-4). For fisheries commodities, FAO uses a fragment of HS-4. In the Harmonized System articles are grouped largely according to the nature of the materials of which they are made, as has been traditional in customs nomenclatures. The HS contains approximately 5000 headings and subheadings covering all articles in trade.

The SITC coding system reflects various aspects of commodities including the materials used in production, the processing stage and the importance of the commodities in terms of world trade. It has a hierarchical structure consisting of Sections, Divisions, Groups, Subgroups and Items. The SITC coding system is available in the following languages: Arabic, Chinese, English, French, Russian, and Spanish. Only the necessary fragment of SITC is used in FAO for fisheries commodities.

4.5 Vessel types and size

In order to assess fleet capacity it is necessary as a bare minimum to have estimates of vessel numbers and main vessel characteristics, such as the vessel type and its size or length.

In international law, as well as in practice, several systems of tonnage measurement have existed side by side. Traditionally, records of measurements of a ship's size were expressed in tons of 100 cubic feet each called Gross Register Tonnage (GRT), as defined by the Oslo Convention (1947). Tonnage was used as a basis for taxes, berthing, docking, and passage through canals and other facilities. However, the method of tonnage measurement has evolved and differs considerably from country to country. A number of international meetings on the subject concluded with the International Convention on Tonnage Measurement of Ships (London, 1969). The Convention, commonly known as the 1969 Tonnage Convention, entered into force in July 1982, though existing ships were not required to comply with the Convention until July 1994. At that time, Gross Tonnage (GT) as defined by the 1969 London Convention became obligatory for all vessels of 24 metres in length and over engaged in international voyages.

Although the London Convention has been adopted for vessels of 24 metres in length and over, for many vessels only data conforming to the Oslo Convention are available. The situation varies from country to country.[7]

Based on the international convention in use, FAO fleet data on the vessel tonnage are measured according to the Oslo Convention (1947) expressing data by GRT [ISSCFVgrt] until 1995; and according to the London Convention (1969) expressing data in GT since 1996 [GT]. As for the type of vessels, the International Standard Statistical Classification of Fishery Vessels by Vessel Types (ISSCFV), based on the type of gear used by the vessels, approved by the CWP in 1984 is adopted [ISSCFVgrt].

Starting with the collection of data for 1996 several other changes were implemented in the form used to gather data: non-fishing vessels were excluded from the inquiry, numbers and capacity data are now collected for broad groups of fishing vessel types and length has been defined as the main characteristic of measurement in international data collation. Discussions are ongoing within the CWP on the possibility of further improvements to the ISSCFV classification "by type" to reflect the state of current technology developments.

[7] The two conventions produce very different tonnage values. Although GT measurement is higher than GRT, there is no simple correlation between the two units (GT is often double the GRT, but sometimes as much as four times the GRT).

4.6 Gear types

The type of gear installed on a vessel determines the type of fish that it can catch, therefore it is often used in statistical collection to determine the fleet power. The main classification of gear types is the International Standard Statistical Classification of Fishing Gear (ISSCFG), adopted in 1980 during the 10th Session of the CWP [ISSCFG]. Although this classification was initially designed to improve the compilation of harmonised catch and effort data questionnaires and in fish stock assessment exercises, it has also been found to be very useful for fisheries technology and the training of fishermen. It has been used in particular for reference in works dealing with the theory and construction of gear and for the preparation of specialized catalogues on artisanal and industrial fishing methods. The classification of gear is used in FAO only for the compilation of fishery factsheets.

5 Creation and population of ontologies from the FIGIS database

In this section we briefly describe the structure of the database in which the reference data is stored in order to point out the salient features that need to be taken into consideration when populating ontologies with data coming from such database.

5.1 The FIGIS database

The main idea underlying the database of reference tables is that anything is an *item* that can have information attached. For example, a territory is an item (endowed with one or more names, codes according to various international coding systems and so on) that can be represented as a row in a table. Similarly, a continent is also thought of as an individual (also endowed with names and codes) represented as a row in the same table. According to the same scheme, all elements of a biological taxonomy (e.g., species, orders) are considered items.

A flag (called *meta code*) is used to distinguish what *type* of object each item is, e.g., a country, a species, a water subdivision and so on. Meta codes are organized into a strict hierarchy, with a common root (called *figis object* after the name of the database) under which each domain is shaped as a strict sub-hierarchy. For example, all domains described in the previous section correspond to a sub-hierarchy starting at the first level of the main hierarchy. Table 3 presents a fragment of the FIGIS hierarchy of meta codes.

1 figis object
- 10 000 Land area
  - 11 000 Geographical region
    - **11 002** Continents
  - 12 000 Groups and union of countries
    - **12 001** Economic unions
  - 13 000 Country, political or statistical entity
    - **13 001** Country
- 20 000 Water area
  - 21 000 Environmental area
    - **21 001** Inland/Marine
  - 22 000 Fishing Statistical area
    - 22 001 FAO statistical area
    - **22 010** FAO major fishing area
      - **22 020** Subarea
      - **22 030** Division
    - **22 040** Subdivision
- 30 000 Biological entity
  - 31 000 Taxonomic entity
    - **30 001** Group
    - **30 002** Order
    - **30 003** Family
    - **30 005** Species
- 40 000 Fishery commodities
  - **40 001** Commodity (FAO ISSCFC classification)
  - **40 002** Commodity (Harmonized classification)
- 50 000 Gear type
  - 51 000 International
    - **54 000** Gear category
    - **54 001** Gear subcategory
  - **54005** ISSCFG
- 60 000 Vessel size categories
  - 61 000 Vessel length classification
    - **61 010** Vessel length class
  - 62 000 Vessel GRT classification
    - **62 020** Vessel GRT division
- 64 000 Vessel type
  - **64 200** Vessel category

Table 3. A fragment of the hierarchy of meta codes (those used are in bold).

The entire hierarchy of meta codes is stored in one *meta table* (called md_refobject in the database). Data concerning each domain is then organized into two tables: [8]

1. one *item table*, where all items in the domain are listed, together with all pieces of information attached to them (e.g., names, codes, meta etc.), and
2. one *group table*, in which the actual hierarchy is stored. The group table is a (four-column) table that renders any hierarchy as a group-member structure. It contains the ID from the item table of both group and member (foreign keys), plus the meta code of the group.

Note that there is only one meta table in the database, so all item tables and group tables refer to it by means of foreign keys. All hierarchies within a domain can then be unrolled by looking at a total of three tables: the meta table, and the item and group tables corresponding at the domain at hand. For example, in order to get and interpret all reference data concerning biological species (cf. Section 4.3) one needs to look at the meta table md_refobject, at the item table called fic_item, and at the group table called fic_item_grp (Figure 4).

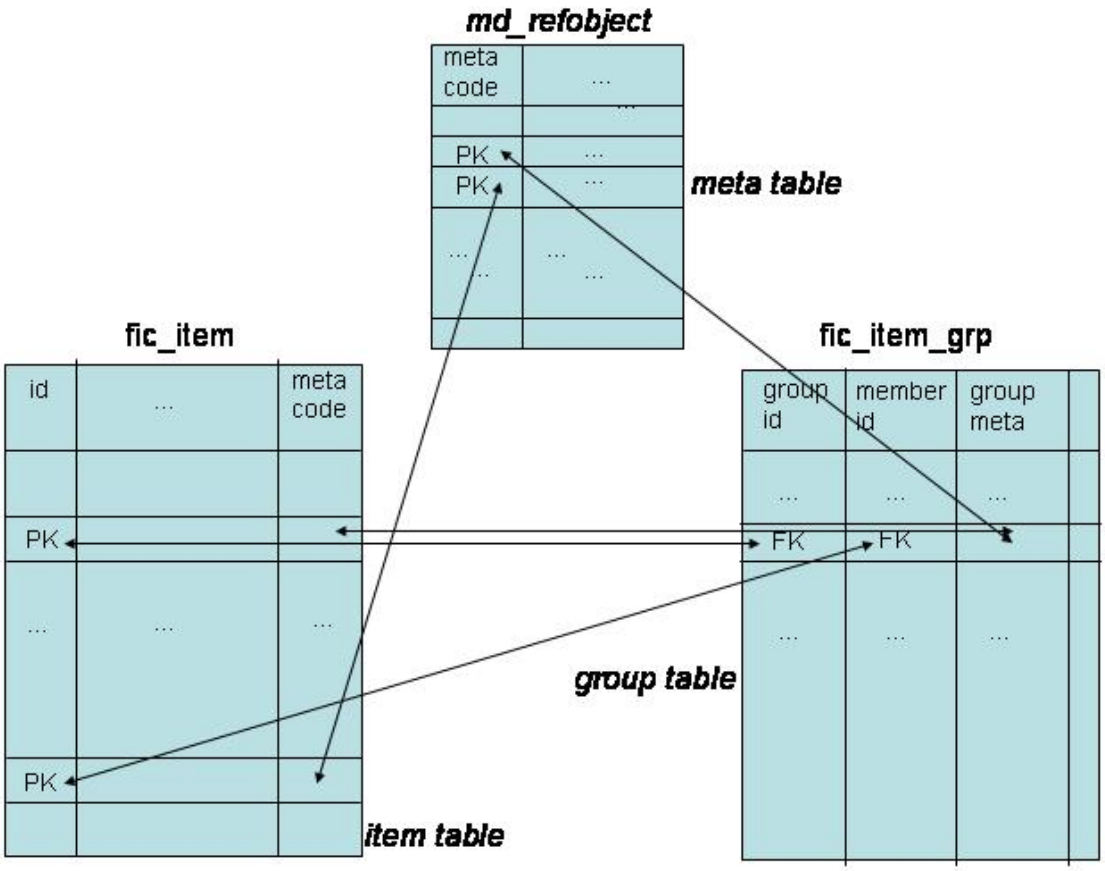


Figure 4. The FIGIS database: tables for the domain of biological entities.

In some cases, a special type of meta codes, called *filters*, are used. The only difference between a meta code and a filter is the following. A meta code is associated with each item in the database, therefore meta codes appear in the three tables mentioned above. A filter is a meta code that is not

[8] Note that there is no table in the database called *meta table*, *item table* or *group table.* This terminology is only used to help the reader grasp the high level structure of the database.

associated with any item. Filters are only used to create hierarchies, and therefore they appear in the meta table and in the group tables, but not in the item tables.

Since the group table only contains pairs of codes corresponding to group-member association, in order to reconstruct any hierarchy deeper than two levels it is necessary to apply self joins (Figure 5).

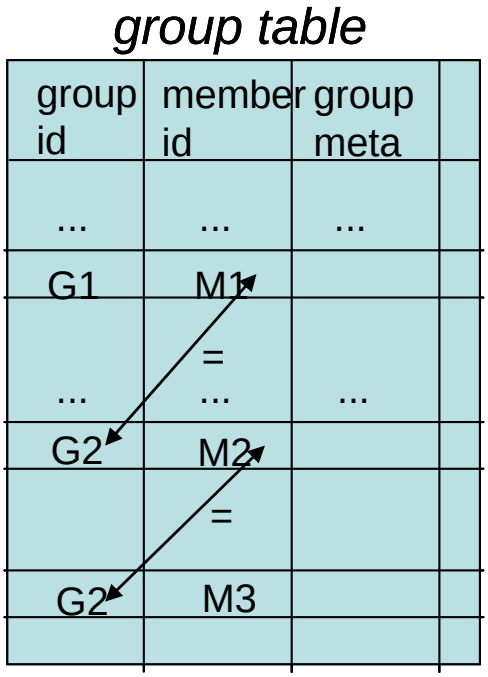


Figure 5. Typical structure of a group table, where an element appears both as group and member (e.g., M1=G2).

Moreover, in order to be able to associate the information stored in the item table with the hierarchical information stored in the group table, it is necessary to apply left and inner joins.

5.2 Population of ontologies from database

Various techniques exist for populating ontologies from existing databases, including [BIZ03, BAR03, PER05]. For our task we selected ODEMapster [BAR06,BAR07], a tool currently under development at the Universidad Politecnica de Madrid.[9]

ODEMapster is an engine that executes mappings between an ontology model and a database by means of a declarative language, R2O. R2O allows the description of complex mapping expressions between ontology elements (concepts, attributes and relations) and relational elements (relations and attributes). It is based on conditions and operations and on rule-style mapping definition for attributes. R2O is independent of the particular RDBMS used.

The ODEMapster Processor generates ontology instances from relational instances based on the mapping description expressed in an R2O document. It can operate at run-time (on-demand query translation) or it can perform massive batch process that generates all possible ontology individuals from the data repository. The operations of ODEMapster are not limited by the

[9] The NeOn toolkit currently allows one to query a database on the basis of an ontology, but does not allow the export of entire data sets according to an ontology model.

expressivity of the DBMS. The set of primitives can be extended with delegable or non delegable primitive conditions and operations. The processor will delegate the execution of certain actions to the RDBMS and execute the rest by itself (post processing).

The main steps of its executions are: Query and R2O parsing, SQL generation, RDBMS execution result grouping and finally post-processing.

5.3 Iteration of conceptualization and population

Since the data at hand is stored in relational form the process of converting it into ontologies is at the same time a problem of domain modelling and data reengineering. Below we describe the two main phases of the process we followed, namely domain conceptualization and the actual population of the ontologies, and highlight the iterative nature of this process (cf. step 1, Figure 1, Chapter 1).

5.3.1 Conceptualization

In order to obtain an adequate knowledge of the domain covered by the reference data, we studied all the available material (bibliographic references are provided in Chapter 4), including the relevant fact sheets from the Handbook of Fishery Statistical Standards by the Coordinating Working Party on Fishery Statistics (CWP), and the actual classification systems used. After having obtained a general overview of the domain, we interviewed domain experts who gave us a practical understanding of the rationale behind the adopted classification systems and of the connections between the reference data and the statistical data collected,

The first models for the ontologies based on reference data were created only by looking at the domain, as explained above. We used Protege 3.1.1 to create and edit the ontology models (on the basis of common metholodogies and best practices for ontology creation, such as [ONTO101]). In many cases we created two alternative models for each domain: the main differences between these models concerned the modelling of codes and names in various languages (i.e., datatype properties vs object properties) and that of hierarchies (e.g., subclasses of biological entities, and part-of hierarchies of land areas). We wrote the corresponding documentation in which we analyzed the pros and cons of each choice. It was decided that the preference for one model over another should be determined not only on the basis of a sound modelling, but also of the efficiency in actual use of the ontology, and of the efficiency in getting the data to populate the ontology.

5.3.2 Population

In order to analyze the FIGIS database we integrated the study of the available documentation (cf. Section 5.1) with a number of interviews with the information experts working with the database. From these interviews we obtained a deeper understanding of the FIGIS database and of the modelling choices it implements. They also gave us insight into the lifecycle of the reference data in the context of real applications and actual use.

From the analysis of the database we found that different flavors of hierarchies are present and encoded in a similar way (cf. Section 7.3). We also found that often only a subset of the available classification system is used, as in the case of biological entities, which taxonomy does not include the Genus (because no timeseries are available for that), or as in the case of fisheries commodities. Fisheries commodities are also an example of hierarchical classifications that are

stored in a non-hierarchical way in the database. These findings made us adjust the ontology models in order to accommodate the specificities of the available data.

5.3.3 Iteration of modelling and population

We used ODEMapster (cf. Section 5.2) to automatically populate the ontology. The ontologies presented in this deliverable are the result of a number of iterations of the cycle conceptualization-population. In the course of these iterations, a number of modelling decisions were made (for a detailed discussion see Chapter 7) for the twofold reason of improving the quality of the models and making them suitable for use together with ODEMapster to extract data from the database. This was the case, for example, for the decision between datatype and object properties, being the former type being easier to extract than the latter.

6 Ontology models

Based on the data described in Chapter 4 and 5, we produced the following ontologies:

1. Land areas (Section 6.1),
2. Fishing areas (Section 6.2),
3. Biological entities (Section 6.3),
4. Fisheries commodities (Section 6.4),
5. Vessel types and size (Section 6.5),
6. Gear types (Section 6.6).

The corresponding ontologies (both as model only and including the actual data) are all publicly available from the FAO website at the following URL: http://www.fao.org/aims/aos/fi. Exact addresses are given together with the ontology description in the course of this section.

Names of ontology elements (except instances) are based on English (sometimes shortened). They apply the naming conventions reported in Appendix I, which are based on FAO internal naming conventions. In some cases, our naming convention differs from the one proposed in deliverable D1.1.1 [D7.1.1] (i.e., for the naming of classes and instances, for which we use lower cases and underscores), but it is not incompatible with it. The rationale of the naming convention we adopted is based on smoothing the connection with the underlying database where names are in lower case and with underscores to separate words. This choice should also be seen in light of the envisaged way to access the data in the database: instead of extracting the data from the database and writing it to a file, it will reside in the database and be accessed at run-time.

We also adopt a somewhat non-standard naming of datatype properties, in that besides using the xml:lang attribute we also add the two-character ISO code of the language in the name of the properties. Once again, this is done to comply with a widely used FAO convention, and to facilitate the reading of the ontologies by human users (be they ontology engineers and editors, or software developers).

Names of instances are generated by combining meta and database ID for each instance in the database. This is done in order to ensure uniqueness of the instance names (across the entire set of ontologies) as commonly done when dealing with the FIGIS database. Meta codes and IDs are also rendered (individually) as datatype properties in order to stay aligned with current practices of interaction with the database.

All datatypes are derived from the database with no modifications, in order to preserve as much as possible the ability to interact with the database and with any other systems built to access it.

All ontologies are in OWL xml-rdf.

6.1 Land areas

Populated ontology: (300Kb) http://www.fao.org/aims/aos/fi/land_v1.0.owl

Model only: http://www.fao.org/aims/aos/fi/land_v1.0_model.owl

Classes (instances):

+ group
  - economic_group (25)
  - geographic_group (8)
+ territory (256)

Disjoint classes:

all

Class restrictions:

economic: *forall* hasMember territory

geographic: *forall* hasMember territory

Datatype properties:

1. hasMeta *Domain: group, territory. Datatype: string (functional)*
2. hasID *Domain: group, territory. Datatype: string (functional)*
3. hasNameShort *Domain: group, territory. Datatype: string*
   a. hasNameShortEN *xml:lang=en (functional)*
   b. hasNameShortES *xml:lang=es (functional)*
   c. hasNameShortFR *xml:lang=fr (functional)*
4. hasNameOfficial *Domain: group, territory. Datatype: string*
   a. hasNameOffEN *xml:lang=en (functional)*
   b. hasNameOffES *xml:lang=es (functional)*
   c. hasNameOffFR *xml:lang=fr (functional)*
5. hasCode *Domain: group, territory. Datatype: string*
   a. hasCodeISO3 *(functional)*
   b. hasCodeISO2 *(functional)*
   c. hasCodeUNDP *(functional)*
   d. hasCodeUN49 *(functional)*
6. hasCoordinate *Domain: territory. Datatype: decimal*
   a. hasMinLat *(functional)*
   b. hasMinLon *(functional)*
   c. hasMaxLat *(functional)*
   d. hasMaxLon *(functional)*
7. hasAreaSize *Domain: territory. Datatype: string (functional)*

8. isValidFrom *Domain: group, territory. Datatype: string (functional)*
9. isValidUntil *Domain: group, territory. Datatype: string (functional)*

Object properties:

1. isInGroup <-> hasMember *Domain: group, territory.*

6.2 Fishing areas

Populated ontology: (130Kb) http://www.fao.org/aims/aos/fi/fishing_areas_v1.0.owl

Model only: http://www.fao.org/aims/aos/fi/fishing_areas_v1.0_model.owl

Classes (instances):

+ fishing_area
  - area (28)
  - subarea (67)
  - division (29)
  - subdivision (10)

Disjoint classes:

all subclasses of fishing_area

Class restrictions:

area: *forall contains* subarea

subarea: *forall contains* division

division: *forall contains* subdivision

Datatype properties:

1. hasMeta — *Domain: fishing_area. Datatype: string (functional)*
2. hasID — *Domain: fishing_area. Datatype: string (functional)*
3. hasName — *Domain: fishing_area. Datatype: string*
   a. hasNameEN — *xml:lang=en (functional)*
   b. hasNameFR — *xml:lang=fr (functional)*
4. hasCoordinate — *Domain: fishing_area. Datatype: decimal*
   a. hasMaxLat *(functional)*
   b. hasMinLat *(functional)*
   c. hasMaxLong *(functional)*
   d. hasMinLong *(functional)*
5. hasAreaSize — *Domain: fishing_area. Datatype: int (functional)*
6. isInland — *Domain: fishing_area. Datatype: Boolean (functional)*

Object properties:

contains — *Domain: fishing_area. Range fishing_area*

6.3 Biological entities

Populated ontology: (13.5Mb) http://www.fao.org/aims/aos/fi/species_v1.0.owl

Model only: http://www.fao.org/aims/aos/fi/species_v1.0_model.owl

Classes (instances):

+ biological_entity
  - group (7)
  - order (112)
  - family (848)
  - species (10604)

Disjoint classes:

all subclasses of biological_entity

Class restrictions:

group: *forall includesOrder* order, *forall includesFamily* family, *forall includesSpecies* species

order: f*orall includesfamily* family, *forall includesSpecies* species

family: *forall includesSpecies* species

Datatype properties:

1. hasMeta *Domain: biological_entity. Datatype: string (functional)*
2. hasID *Domain: biological_entity. Datatype: string (functional)*
3. hasName *Domain: biological_entity. Datatype: string*
   a. hasNameEN *xml:lang=en (functional)*
   b. hasNameES *xml:lang=es (functional)*
   c. hasNameFR *xml:lang=fr (functional)*
4. hasNameFull *Domain: biological_entity. Datatype: string*
   a. hasNameFullEN *xml:lang=en (functional)*
   b. hasNameFullES *xml:lang=es (functional)*
   c. hasNameFullFR *xml:lang=fr (functional)*
5. hasNameLong *Domain: biological_entity. Datatype: string*
   a. hasNameLongEN *xml:lang=en (functional)*
   b. hasNameLongES *xml:lang=es (functional)*
   c. hasNameLongFR *xml:lang=fr (functional)*
6. hasNameScientific *Domain: biological_entity. Datatype: string (functional)*
7. hasCode *Domain: biological_entity. Datatype: string*
   a. hasCodeTax *(functional)*
   b. hasCodeAlpha3 *(functional)*

Obect properties:

1. includesOrder *Domain: biological_entity, range: biological entity*
2. includesFamily *Domain: biological_entity, range: biological entity*
3. includesSpecies *Domain: biological_entity, range: biological entity*

6.4 Fisheries commodities

Populated ontology: (10.6Mb) http://www.fao.org/aims/aos/fi/commodities_v1.0.owl

Model only: http://www.fao.org/aims/aos/fi/commodities_v1.0_model.owl

Classes (instances):

+ fi_commodity
  - isscfc (1230)
  - hs (140)

Disjoint classes:

all subclasses of fi_commodity

Class restrictions:

hc: *forall hasCorrISSCFC* isscfc

Datatype properties:

1. hasMeta *Domain: fi_commodity. Datatype: string (functional)*
2. hasID *Domain: fi_commodity. Datatype: string (functional)*
3. hasName *Domain: fi_commodity. Datatype: string*
   a. hasNameEN *xml:lang=en (functional)*
   b. hasNameES *xml:lang=es (functional)*
   c. hasNameFR *xml:lang=fr (functional)*
4. hasNameLong *Domain: fi_commodity. Datatype: string*
   a. hasNameLongEN *xml:lang=en (functional)*
   b. hasNameLongES *xml:lang=es (functional)*
   c. hasNameLongFR *xml:lang=fr (functional)*
5. hasNameFull *Domain: fi_commodity. Datatype: string*
   a. hasNameFullEN *xml:lang=en (functional)*
   b. hasNameFullES *xml:lang=es (functional)*
   c. hasNameFullFR *xml:lang=fr (functional)*
6. hasCode *Domain: fi_commodity. Datatype: string*
   a. hasCodeISSCFC *(functional)*
   b. hasCodeHS *(functional)*
   c. hasCodeSITC *(functional)*

Object properties:

1. hasCorrISSCFC *Domain: fi_commodity,*
   Range: fi_commodity

6.5 Vessel types and size

Populated ontology: (100Kb) http://www.fao.org/aims/aos/fi/vessels_v1.0.owl

Model only: http://www.fao.org/aims/aos/fi/vessels_v1.0_model.owl

Classes (instances):

+ vessel type
  + by_length
    - grt (12)
    - gt (15)
  - by_type (93)

Disjoint classes:

all

Class restrictions:

none

Datatype properties:

2. hasMeta — *Domain: vessel_type. Datatype: string (functional)*
3. hasID — *Domain: vessel_ type. Datatype: string (functional)*
4. hasName — *Domain: vessel_ type. Datatype: string*
   a. hasNameEN — *xml:lang=en (functional)*
   b. hasNameES — *xml:lang=es (functional)*
   c. hasNameFR — *xml:lang=fr (functional)*
5. hasCode — *Domain: vessel_ type. Datatype: string*
   a. hasCodeFAO *(functional)*
   b. hasCodeISSCFV *(functional)*
6. hasStdAbb — *Domain: vessel_ type. Datatype: string (functional)*
7. hasDescription — *Domain: vessel_ type. Datatype: string*
   a. hasDescEN — *xml:lang=en (functional)*
   b. hasDescES — *xml:lang=es (functional)*
   c. hasDescFR; — *xml:lang=fr (functional)*
8. hasEntryDate — *Domain: vessel_ type. Datatype: datetime (functional)*
9. hasVessClassGRT — *Domain: vessel_ type. Datatype: string*
10. hasVessClassLength — *Domain: vessel_ type. Datatype: string*
11. hasVessClassPower — *Domain: vessel_ type. Datatype: string*
12. hasLowerLimit — *Domain: by_length. Datatype: string*
13. hasUpperLimit — *Domain: by_length. Datatype: string*

6.6 Gear types

Populated ontology: (80Kb) http://www.fao.org/aims/aos/fi/gears_v1.0.owl

Model only: http://www.fao.org/aims/aos/fi/gears_v1.0_model.owl

Classes (instances):

+ isscfg
  - level1
  - level2
  - level3

Disjoint classes:

all

Class restrictions:

none

Datatype properties:

1. hasID *Domain: isscfg. Datatype: string (functional)*
2. hasMeta *Domain: isscfg. Datatype: string (functional)*
3. hasName *Domain: isscfg. Datatype: string*
   a. hasNameEN *xml:lang=en (functional)*
   b. hasNameES *xml:lang=es (functional)*
   c. hasNameFR *xml:lang=fr (functional)*
4. hasDescription *Domain: isscfg. Datatype: string*
   a. hasDescEN *xml:lang=en (functional)*
   b. hasDescES *xml:lang=es (functional)*
   c. hasDescFR *xml:lang=fr (functional)*
5. hasEntryDate *Domain: isscfg. Datatype: dateTime (functional)*
6. hasStdAbb *Domain: isscfg. Datatype: string (functional)*
7. hasCodeISSCFG *Domain: isscfg. Datatype: string (functional)*

Object property:

hasSublevel *Domain:isscfg. Range: isscfg*

7 Discussion

All ontologies based on FIGIS consist of a small number of classes, ranging from three (commodities) to five (land areas, water areas, biological entities, vessels), and a medium to large number of instances, ranging from seven (class *group*, in species_v1.0.owl) to 10604 (class *species* in the same ontology).

The modelling style is consistent. The same modelling style if for example adopted in the use of datatype properties for names and codes, in the use of (universal) constraints and in the definition of domain and range of both datatype and object properties.

7.1 Selection of properties

All pieces of information of most common use by the application for fisheries have been included in the ontologies, including the ID of all items in the database and the meta code used to identify the "type" of reference data at hand. Since the combination of ID and meta code is unique within the database we also use this combination to form the names of all instances.

All data included in the ontologies comes from the database, without further additions or modifications. As a consequence, sparsely populated columns in the database are rendered as empty properties. Since this emptiness is to be related to the dynamic nature of the database, we preferred not to impose constraints on required or non-required properties.

Some properties only make sense in conjunction with others, as in the case of geographical coordinates. In the FIGIS database coordinates are given in order to circumscribe areas, so 4 coordinates are usually stored for each area: minimum and maximum latitude and minimum and maximum longitude. In that case it is important to distinguish each coordinate from the other, but also to group all of them together. For this reason we preferred to use one property per coordinate, and to group them as subproperties of the same superproperty (cf. Section 6.1).

7.2 Managing multilinguality

Most of the reference data is available in more than one language, usually English, French and Spanish. Often, two or three names are available in each language, such as a "short name," a "long name" and an "official name." In all cases, names are established on the basis of international agreements that result in a 1-to-1 correspondence between languages.

We rendered each name (either short, long or official) by means of datatype properties, endowed with an rdf label *xml:lang* corresponding to the language at hand. The two-digit ISO codes of the language are also kept in the name of properties (as in "hasNameShortFR") in order to ease the visualization of properties by human editors and to stay close to existing practices adopted in FAO.

Given the sharp 1-1 correspondence between languages, the ontologies based on FIGIS constitute a simpler case of management of multilinguality than the model that is being proposed withing NeOn.[10]

[10] The NeOn model for multilinguality will described in deliverable D2.4.1, due at month 18, in parallel with the present deliverable.

7.3 Different flavours of hierarchies

The FIGIS database is organized according to a hierarchical structure (cf. Section 5.1). Also, most of the coding systems are taxonomic, as in the case of the “taxonomic code” for biological items, FAO divisions for water areas, HS and ISSCFC for commodities, ISSCFV for vessels, ISSCFG for gears. However, not all data with taxonomic codes is stored in the same way.

In the case of FAO water areas, there is a strict correspondence between the FAO taxonomic code and the hierarchy encoded in the group table. This means that the three pairs: major area, subarea; subarea, division; and division, subdivision are present in the group table as pairs of group and member (see Figure 5). Moreover, since this is a hierarchy based on physical inclusion, it is always complete (an area may not have subareas, but all subareas have an area to which they belong).

Biological items have taxonomic codes, but they do not strictly correspond to the hierarchy stored in the group table. In fact, taxonomic codes include a place for the genus, which is not actually used in the statistical database. Moreover, the chain from main group to species may not be complete, as there are species that have no family or order in the database, but only a main group. This is the reason why we used three object properties (includesOrder, includesFamily, includesSpecies) instead of only one (as in the case of the fishing water areas).

Also fisheries commodities have taxonomic codes associated (ISSCFC and HS) but the taxonomy expressed by these codes are not encoded in the group table. That table only represents the *correspondence* between each commodity item in the HS and items in the ISSCFC system. In other words, in that case the parent child structure is used not to express a hierarchy *within* a coding system but to express the relationship between two different systems. This is the reason why commodities are represented in the ontology as two flat lists of instances, despite their complex taxonomic codes.

Finally, the main classification of gear types is based on a filter, not on a meta code. This means that ISSCFG objects do not exist as such in the item table, but can only be reconstructed by looking at the ISSCFG filter code and at the corresponding entries in the parent table.

7.4 Mapping

The creation of mappings between ontologies is out of the scope of this deliverable. However, the ontology of commodities is an example of an ontology that could be managed as two ontologies linked by mapping. In fact, the two classifications for commodities used are maintained by different organizations: the ISSCFC is maintained by FAO, and the HS is maintained by the World Customs Organization (WCO). It could be more effective and safer to manage them in two different ontologies. In order for this strategy to be applied, a sound mechanism for editing, maintaining, and versioning mappings should be in place. Special attention should be paid to the handling of changes undergone by the ontologies to be linked.

Also, some of the ontologies produced can be easily put in “contact’ to one another. Consider for example the relation between land areas and water areas, or the relation between commodities and biological species. These relations would be best represented by means of mappings across ontologies instead of relations within the same ontologies.

Finally, much information highly related to the domains at hand could be taken from other sources (other databases, text documents, human knowledge) and would increase enormously the potential of the ontologies produced. These pieces of information could be added within single ontologies (for example borders of land areas, territories and groups), or they could “link” two ontologies (for example about shores, thus involving territories and water areas). Also linguistic information such as names of territories and commodities in languages not included in the reference tables could be added by some form of mapping.

8 Lessons learned

In this chapter we summarize the lessons learned during the generation of ontologies based on FIGIS.

8.1 Using non-integrated tools is error prone and time consuming

In order to populate the ontologies from the database, we followed the steps listed in Table 4. Numbers in the first column correspond to the sequential order for the action, which is listed in the second column, while the tools we used are listed in the third column.

Step	Action	Tool used
1	Inspect and query the DB	MySql or Oracle client, Navicat, Access
2	Edit and visualize the ontology model	Protégé, NeOn toolkit
3	Edit the R2O and query files	XMLSpy
4	Run the ODEMapster processor	Windows command prompt
5	Generate a single file with model and instances	Shell on Windows environment (cygwin)
6	Inspect the ontology (class definition and instances) as a whole	Protégé, NeOn toolkit, Semantic Networks

Table 4. Steps followed for the creation and population of the fisheries ontologies.

Often one or more steps in the sequence had to be iterated; and each step involved the use of a specific tool. In particular, steps 1-2, 2-3 and 3-4-5-6 are often repeated as groups.

We found that the use of different tools, totally independent and not integrated into a single environment, is time consuming and cumbersome to use. In particular, we run into the following issues:

1. Use of different language encoding. FIGIS uses UTF-8 by default, but in other applications this is not the case and/or it may be difficult to choose the desired encoding.
2. Selection of the right datatype to assign to ontology properties. The same datatypes used in the FIGIS database should be assigned to the ontology properties. In order to do this, it is necessary to inspect the target database in a seamless manner, and visualize at least the table descriptions and thier datatypes.
3. Use of different flavours of OWL dialects. It happens that different tools (e.g., Protégé and NeOn toolkit) support different fragments of OWL: this fact determines what operations can be performed with a tool and what can be saved.
4. Different default namespaces (some cumbersome editing was necessary in order to harmonize the namespaces manually given at the time of the creation of the ontology model, and the namespace automatically created by ODEMapster).
5. The window-based tools, including Protégé and the NeOn toolkit, had problems with opening and manipulating large ontologies (files) of that size. For this reasons, we largely used shell commands to inspect and edit the instances generated by ODEMapster and also the ontologies including both instances and model. This is then one more reason for

accessing the data directly from the database and putting the semantic layer on top of it, and to develop efficient mechanisms for managing ontology modules (cf., Chapter 7).

At least the two functionalities of ontology editing/visualization and the creation of the schema to query the DB and populate the ontologies should be integrated in a single environment.

8.2 Self-joins are critical to working with FIGIS

The process of accessing data from the database and converting it into an ontological model was found to be a complex task. Such complexity was not due to the number of classes to define (small), nor from their definition or the number of instances available for each class, but from the way hierarchies are stored in the database. In fact, as we described in Chapter 5, the reference tables are organized in the database as a "family" of hierarchies, where the structure of the main hierarchy is stored in the meta table and each individual domain hierarchy is stored in a separate group table. Then, in all those cases in which the domain hierarchy is deeper than two levels (i.e., the first level of parent-child relation), it is necessary to apply a self-join.

Since the version of ODEMapster that we used for our work did not support self-joins, we had to pre-process the tables and create new areas where the hierarchy is unrolled. This pre-processing is not applicable in a realistic scenario (it clashes with our requirements concerning maintenance and updating of the data) and it is both time consuming and error prone. Based on the feedback provided during this work, a new version of ODEMapster is under development, one which addresses this specific problem and in so doing it will be able to deal with structures such as the one used by the RMTS.

8.3 Graphical interfaces are critical, but they should also be flexible

The graphical interface (GUI) available for ODEMapster was rather basic, so we manually wrote the R2O schema and the query files. We found that XMLSpy [XMLSpy] was the tool most useful for the task (even though the files are not valid XML files). Manual editing of the files had the advantage of giving us the maximum degree of flexibility but it also made the process time consuming and very error-prone. Therefore, an appropriate GUI for the task should allow the user to compose schema and query files without actually having to know all the syntax details of the language. At the same time the experienced user should also be allowed to inspect and edit the code directly. In that case, appropriate documentation based on examples, syntax specification and tutorials should be available. A further improvement of the tool would consist in a mechanism to automatically generate the mapping to the database from a given ontology model. Finally, special attention should be paid to the debugging environment, which should be improved in order to clearly highlight the nature of the error and the place in the code where the error is located.

Finally, when legacy data is left in databases and accessed and queried by ontologies, it would extremely useful to be able to use the ontology as an interface for editing the data residing in the database. Once again, this implies that the visualization tools be flexible and adaptable enough to support this type of operation, and that permissions, versions and backup be correctly handled.

8.4 If efficiency is an issue, modularization is required

Some of the ontologies created are rather large (e.g., biological entities is 13.5 Mb), which causes problems when loading them into most software (see Chapter 8) and when operating with them. Some notions of "modularization" is required in order select and load only the portion of ontologies that need to be visualized and accessed. Given the fact that the ontologies based on reference tables are richer in instances than in classes, a convenient solution would be to select only the

specific sets of properties, for example only one or more languages or only one or more classification codes out of the many available. Appropriate facilities for visualization, editing, and versioning these modules should be available.

9 Conclusions and next steps

In this deliverable we presented (Chapter 6) and discussed (Chapter 7) the first set of fisheries ontologies created within WP7, that are available at the following URL: http://www.fao.org/aims/aos/fi. All ontologies are based on fisheries reference tables. We populated the ontologies with data stored in a relational database. This allowed us to verify that it is possible to extract data from the database according to an ontological model, but not all the machinery needed is currently in place. In order to allow the NeOn partners to overcome this problem, we described in detail the structure of the database at hand and the problems we ran across during our work. We also listed the lessons learned during this exercise (Chapter 8), which will be useful to the advancement of the project.

We succeeded in creating the desired ontologies from the database but this was only possible by pre-processing the data. This solution is obviously not sustainable in a real setting, because of the work involved and because in practice it would make the ontologies non-updatable and therefore non usable.

Future work includes the creation of additional ontologies, including also ASFA and AGROVOC. This work will be reported on in Deliverable D7.2.3 (Enhanced Networked Fishery Ontology) due at M30. We will proceed by incrementally adding ontologies based on the inventory [D7.2.1], connecting them to one another to form a network, and experimenting with adding relations not present in the FIGIS database but available from other sources and with adding relations across the ontologies created so far. In order to do this, appropriate mapping mechanisms and tools for editing, visualizing, storing and versioning mappings should be in place. We listed possible examples of these mappings in Chapter 7.

Annex I. Naming conventions

URI based: www.fao.org/aims/aos/fi/

Ontology names: lower letters, words separated by underscore. The name should include the version number, in the form: "_vx.x", just before the entension.

Example: name_ontology_version.owl

Classes: lower letters, with underscore instead of spaces.

Example: http://www.fao.org/aims/aos/fi/

Properties and Relations: Camel style.

Example: hasMeta, hasCodeISO2, hasNameEN.

Instances: combination of meta code and id from the database. For example: "54002_ 105".

Annex II. Glossary of fisheries terms

Catch The total number (or weight) of fish caught by fishing operations. Catch should include all fish killed by the act of fishing, not just those landed. The catch is usually expressed in terms of wet weight. It refers sometimes to the total amount caught, and sometimes only to the amount landed. The catches which are not landed are called discards.

Commodity Goods and services which are the result of production processes normally intended for sale on the market at a price that is designed to cover their costs of production.

Fishery Generally, a fishery is an activity leading to harvesting of fish. It may involve capture of wild fish or raising of fish through aquaculture. A fishery can also be taken as a unit determined by an authority or other entity that is engaged in raising and/or harvesting fish. Typically, the unit is defined in terms of some or all of the following: people involved, species or type of fish, area of water or seabed, method of fishing, class of boats and purpose of the activities.

Fishery Fleet The term "fishery fleet" or "fishery vessels" refers to mobile floating objects of any kind and size, operating in freshwater, brackish water and marine waters which are used for catching, harvesting, searching, transporting, landing, preserving and/or processing fish, shellfish and other aquatic organisms, residues and plants.

Fishing Vessel The term "fishing vessel" is used instead when the vessel is engaged only in catching operations.

Non-Fishing Vessel The term "non-fishing vessel" applies to vessels performing other functions related to fisheries, such as supplying, protecting, rendering assistance or conducting research or training.

Gear A fishing gear is a tool used to catch fish, such as hook and line, trawl, gill net, trap, spear, etc.

Gross Register Tonnage (GRT) The Gross Register Tonnage represented the total measured cubic content of the permanently enclosed spaces of a vessel, with some allowances or deductions for exempt spaces such as living quarters (1 gross register ton = 100 cubic feet = 2.83 cubic metres).

Gross Tonnage (GT) The Gross Tonnage for ships of 24 metres in length and over refers to the volume of all ship's enclosed spaces (from keel to funnel) measured to the outside of the hull framing.

Inland Water The surface water existing inland, including lakes, ponds, streams, rivers, natural or artificial watercourses and reservoirs, and coastal lagoons and artificial water bodies.

Nominal Catch The sum of the catches that are landed (expressed as live weight equivalent). Nominal catches do not include unreported discards.

Production The total living matter (biomass) produced by a stock through growth and recruitment in a given unit of time (e.g. daily, annual production). The "net production" is the net amount of living matter added to the stock during the time period, after deduction of biomass losses through mortality. Also: The total elaboration of new body substance in a stock in a unit of time, irrespective of whether or not it survives to the end of that time. Also called: net production.

Annex III. List of acronyms

ASFIS Aquatic Sciences and Fisheries Information System

CWP Coordinating Working Party on Fishery Statistics

FIES FAO Fisheries and Aquatic Information and Statistical Service

GRT Gross Registered Tonnage

GT Gross Tonnage

HS Harmonized Commodity Description and Coding System

ISO International Organization for Standardization

ISSCAAP International Standard Statistical Classification of Aquatic Animals and Plants

ISSCFC International Standard Statistical Classification of Fishery Commodities

ISSCFG International Standard Statistical Classification of Fishing Gears

ISSCFV International Standard Statistical Classification of Fishing Vessels

RT Reference Tables

SITC Standard International Trade Classification of the UN

Annex IV. Reengineering the XML schema for FI Factsheets to OWL

Generalities

A large amount of information about fisheries, aquaculture and related subjects, including fishing techniques, fishing areas, fishery and aquaculture country profiles, is available in the form of fact sheets [FS]. All Fisheries fact sheets in FAO are in documents in XML format, structured according to a comprehensive XML schema [FSschema] that includes all elements used in all types of fact sheets.

Fact sheets are organized into domains (e.g., Aquaculture species, Fishing equipment, Fishery, Gear type), each corresponding to an element of the schema, under FIGISdoc, the root of any fact sheet (XML document). Domains are fully specified by means of nested elements. Each element includes a description meant for human use.

The nested elements that specify FIGIS documents (fact sheets) can represent fishery-related content, but also identifiers, values, types of things referred to, etc.

The schema makes use of existing standard element sets such as Dublin Core [DC], Extended Dublin Core [EDC], AGMES [AGMES] and AIDA [AIDA]. It also incorporates wherever possible existing classification schemes such as ISO standards for countries, currencies, languages, and other fisheries-related international classification schemes.

It is important to note that the schema was conceived as a means for editors to create structured documentation, and as such was not created based on a relational or ontological model, but was rather organised following hierarchical document formatting conventions. A dictionary of the elements used in the schema is available online [FSdic].

In this section, we describe the results of a bulk translation of the schema into an OWL model, what are the issues arising from this translation, and what can be done in order to take advantage of the knowledge contained in the fact sheets, without being overloaded with specifications bound to the documental form, rather than to the conceptual form of fishery information.

We firstly describe the generalities of the XSD schema, then the way the conversion to OWL has been performed, the issues for the usability and usefulness of the ontology, and finally the possible solutions for those issues.

Conversion of the FI XSD to OWL

We have translated the fact sheet schema to OWL, and made a preliminary analysis, trying to understand the semantic issues that derive from the translation.

The schema [FSschema] is available from the FI web site: http://www.fao.org/aims/aos/fi/fi.owl

The schema uses the following additional schemas that are common in the metadata community: AGMES [AGMES], AIDA [AIDA], Dublin Core [DC], Dublin Core Terms [DCT].

Using the TopBraid Composer [TBC] tool (XSD2OWL) for translating from XML Schema (XSD) to OWL, we have translated the FS schema into an OWL model. The TopBraid tool assumes a fairly common semantics for translating XSD; in particular, *elements*, e.g. *FisheryArea*, are converted both as owl:Classes, e.g. *Class:FisheryArea*, and as owl:ObjectProperties, e.g. *ObjectProperty:hasFisheryArea*. A trailing “has” is added in order to distinguish classes and properties resp. that are converted from the same element.

The reason for this translation is that XSD elements are associated to other elements by means of operators like *choice*, and of cardinalities. When an association is stated, a *type* is assumed to be filled. When such a type is an XSD datatype (e.g. string, float, etc.), the element can be converted to an owl:DatatypeProperty. When the type is not an XSD datatype, a trailing “type” is added to the name element in order to declare the new type to be applied in that case. E.g. in Figure 1 the FisheryArea element is associated to the FishingGround element with a “FishingGroundType”. This structure is converted in OWL as an owl:ObjectProperty *hasFishingGround*, an owl:Class *FishingGround*, and an owl:Restriction *(hasFishingGround allValuesFrom FishingGround)* that subsumes the owl:Class *FisheryArea* (Figure 2).

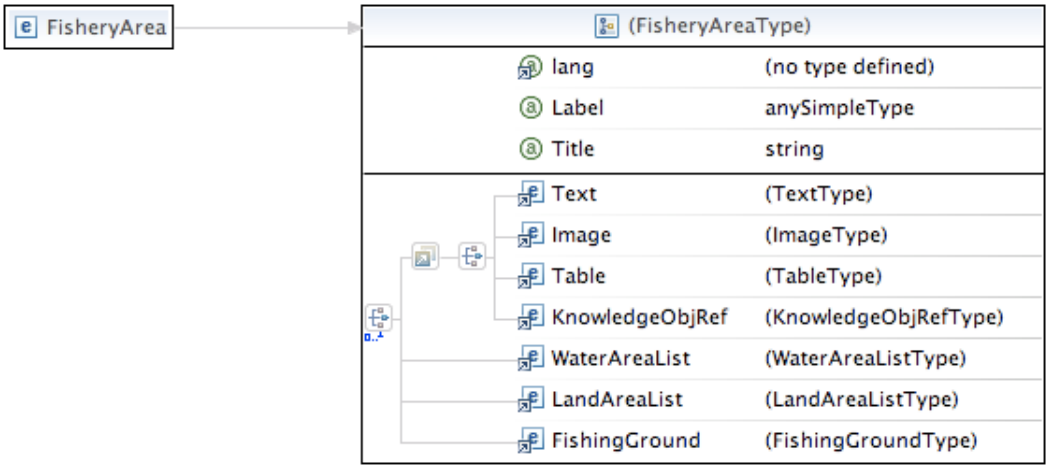


Figure 1. A visualization of the XSD element FisheryArea.

The ontology converted from the schema (fi.owl) is publicly available from the following address: http://www.fao.org/aims/aos/fi/fi.owl.

fi.xsd is made of XSD elements and XSD attributes, and a few XSD groups. No complex types and subtypes have been used. As typical with XSD, names conform to the following pattern:

<element name> <element name[Type]>, where the type has the same name, but with a trailing 'type' attached, e.g.:

FishTechnique --- FishTechniqueType

In fi.owl, this has been converted into the following pattern for OWL naming:

[has]<property name> <class name>, where class and property names are identical, the trailing 'type' has been removed, but the trailing 'has' prevents name clashes in OWL1.0, e.g.:

hasFishTechnique --- FishTechnique

The conversion tool also converts the schemas that are used in fi.xsd (e.g. Dublin Core). Although this conversion can also be interesting, after a discussion with the maintainers of the schema, we have decided to ignore their full conversion into OWL, because we do not need to maintain the conversions of the used schemas as well. Anyway, the elements used by the schema are still present in the converted OWL file with an appropriate namespace. For example, the owl:Class *KnowledgeObjRef* uses the owl:ObjectProperty *hasCreator* translated from the DublinCore Elements namespace (see Figure 4).

The converted fi.owl ontology consists of 653 named classes, 575 object properties, 163 datatype properties, and 15 hybrid properties, which have both object- and datatype-ranges (see issues below). Classes are characterized by means of 8424 occurrences of subsuming restrictions (with an average set of about 13 for each class). Among the occurrences of restrictions, 4695 are universals (allValuesFrom), which translate the integrity constraints from the XSD; 3332 are maxCardinality, 373 are (exact) cardinality, and 24 are minCardinality.

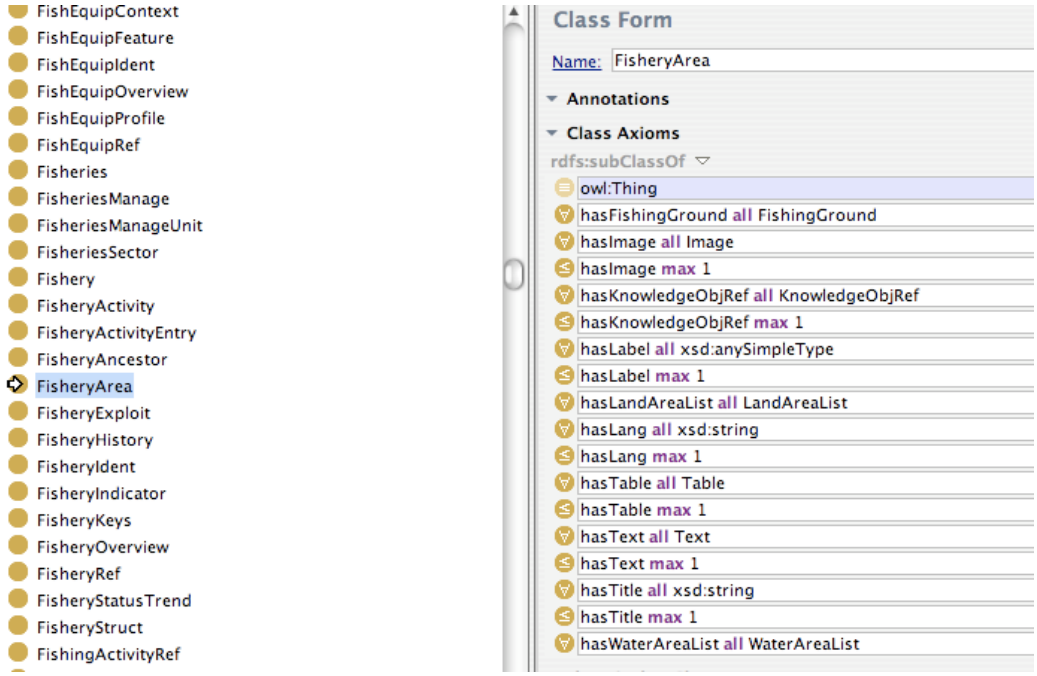


Figure 2. A visualization of the owl:Class FisheryArea, translated from an XSD element.

Evaluating the design of the converted ontology

Converting XSD to OWL is very useful, but the results should be usually reengineered in order to reach an appropriate syntactic and semantic quality that makes the ontology usable and useful. We have carried out an analysis that has made them emerge four issues, which are dealt with solutions in Section 4.

The analysis of the OWL FI schema after the conversion has been annotated by means of owl:versionInfo annotations within the fi.owl file, augmented with a "TODO:" keyword at the beginning of the annotation.

(1) *Compatibility with existing tools and syntactic checks*

fi.owl is loadable on TopBraid Composer and SemanticWorks [SW], while Swoop and Protégé have some syntactic problems, whose cause has not been identitied yet. The NeOn toolkit loads it, but the nominals (oneOf) derived from XSD joins are not visualized, because the NeOn toolkit does not support nominals yet. When loaded, syntactic checks reveal the well-formedness of the ontology.

(2) *Semantic checks and OWL species*

A semantic check has been performed with the Pellet reasoner, and reveals no inconsistencies on the Class hierarchy (quite trivially, since no disjointness axioms or negations are asserted). The presence of 15 hybrid properties causes the ontology to be OWL-Full. At a closer glance, most of the 15 properties can be easily fixed to be either ObjectProperties of DatatypeProperties only. Without this flaw, the ontology is in OWL-DL, due to the many DataRange nominals (about 300), used in order to translate XSD “joins”.

(3) *Design issue I: a hybrid domain of interpretation for the ontology?*

As Figure 3 shows, certain classes refer to textual material or full documents, rather than to objects of the fishery domain. For example, *KnowledgeObjRef* is one of such classes. In fact, when the OWL translation occurs, heterogeneous elements like *FisheryArea*, and *KnowledgeObjRef* become both owl:Classes. Due to the typical conventions in creating OWL ontologies, which suggest to give intuitive names to classes and properties, at a first glance the domain of interpretation of the ontology resulting from the FI schema contains fishery objects (areas, techniques, species, etc.), data, or documents. As said above, this intuition is misleading, because all elements in the FI schema are defined having in mind the management of documents, therefore even those classes that apparently describe fishery objects have to be understood as classes including documental objects that *express* descriptions of fishery objects, or *refer* to fishery objects. This situation is typical in web technology, because html documents can link, refer, or describe either other pages, or real world entities and concepts, by using the same `href` mechanism. In other words, in schemas designed for web applications, it is common to find a mixture of documental and real world knowledge.

Recent research has addressed this issue, known as *identity and reference over the web*, or simplier, as the *identity crisis*. An ontology like IRE (Identity, Reference, and Entities)[11], which is also aligned to the NeOn C-ODO ontology [D211] is able to describe the difference between different web references, between web pages and real world entities, between URIs of ontology elements and URIs of web pages, etc. Based on this analysis, the FI ontology is not homogeneous with the other Fishery ontologies, whose interpretation domain directly address fishery objects.

In the next section, we propose that an appropriate analysis of the documental references in the FI ontology can lead to a reengineering that makes it emerge a more conventional fishery ontology.

(4) *Design issue II: partial consistency in the structure of the documental ontology*

As previously said, the FIGIS XSD has not been defined in order to create a database, or as a typical ontology or conceptual model, but rather in order to maintain a document management system. For example, the notion of “Fish Technique” is spread out into several classes in order to

[11] http://www.loa-cnr.it/ontologies/IRE/IRE.owl

Figure 3. A visualization of the owl:Class *FishTechnique* from fi.owl.

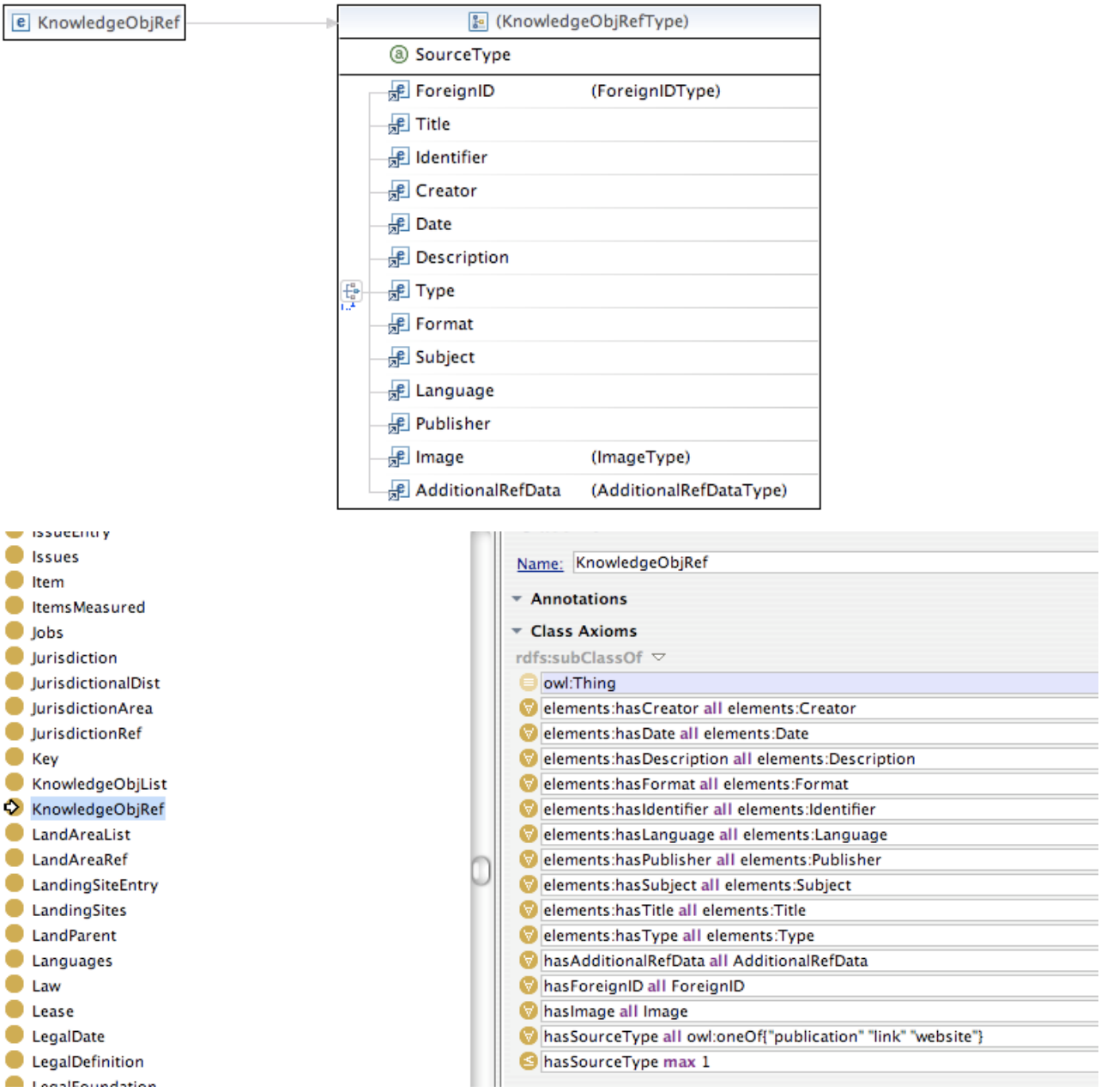
By looking at the attributes and properties of each class (see Figure 3), we can understand the rationale adopted case by case:

- the FishTechnique class is used to collect (directly or indirectly) all the structured information types that are specific to fish techniques: Ident, Ref, Feature, Parent, Profile, Operation, Struct, but also other, non-specific information, such as Season, FisheryArea, etc., and purely documental information, such Biblography, Sources, etc.
- the FishTechniqueIdent class is used to refer to the actual entities (fish techniques), e.g. by relating them to exploitation forms, geartypes, handling modes, species, vessel types, but also images and “parent” techniques (via the FishTechniqueParent class)
- the FishTechniqueRef is used to distill the basic information required to construct a FishTechnique document
- the FishTechniqueOverview class is used to declare the structure of the document, with images, tables, language type, etc.
- etc.

Unfortunately, there is not a stable pattern for encoding different kinds of information into classes with different but related names. For example, with the notion of “Aquatic Resource”, contrary to what is designed for fish techniques, the AqRes class encodes the properties referring mostly to actual entities, while AqResIdent encodes properties of the related information.

As a result, the OWL version of the schema has classes with properties mixing up data about the entities referred in fact sheets (e.g. *FishTechniqueIdent*), data about textual descriptions of those entities (e.g. *KnowledgeObjRef*), and data about the document that expresses textual descriptions (e.g. *Sources*).

Figure 4. A visualization of the XSD element *KnowledgeObjRef*



Dealing with the issues arisen in the evaluation

(1) *Loadability*

Since there is no apparent reason behind the loading errors in Protégé and Swoop (the syntactic check goes plain on Neon Toolkit, TopBraid Composer, and SemanticWorks), we will simply send a note to the maintainers of those tools.

(2) *Hybrid properties*

As said above, the 15 properties causing the ontology to be OWL-Full will be fixed in the next phase of reengineering, and the results and lessons learnt will be reported in the next deliverable. The intervention requires to fix the properties that are used either as owl:ObjectProperty or as owl:DatatypeProperty. For example, *hasStatus* is used with the datatype range *xsd:positiveInteger*, with sets of values (strings), e.g.: *(owl:oneOf{"Received" "Processed" "Screened" "Filled" "Sent"}),* and also with the owl:Class *Status*. This can be due to the lack of a unique name assumption in the development of the schema, or to the tolerance of XSD, which does not check for the consistent use of element names across the schema.

(3) *Hybrid domain of interpretation*

While assuming that the actual domain of interpretation for fi.owl includes only documental objects solves the issue (3), which can be confined to a problem in the naming patterns used for XSD elements, still there is a feeling that the apparently documental knowledge encoded in fi.owl could be partly reused to produce a properly ontology for fishery. In the following we formulate a proposal for an activity that will possibly be reported in the next deliverable.

The proposal basically suggests to consider, in the vein of the “identity crisis”, that relations between documental objects often *mirror* relations between domain objects, in this case fishery objects. In practice, it is useful to assume that the *textual* associations between fishery-related document types that are declared in the FI ontology correspond to *factual* associations between the fishery objects described in a document, e.g. a fishery area associated with a fishing ground. Interestingly enough, it is possible to reengineer the FI ontology, so that it only contains axioms with a strict fishery interpretation, e.g. that the owl:Class *FisheryArea* (once interpreted as a class of geographical areas) has the owl:Restriction *(hasFishingGround allValuesFrom FishingGround)* (interpreted as a class of fishing-relevant places).

This reengineering process needs to remove from the FI ontology all the axioms that refer to strictly documental associations, for example, the owl:Restriction *(hasText allValuesFrom Text)* should be removed in the process.

Other document-oriented associations can be used in different ways, in order to reengineer other kinds of knowledge contained in fact sheets. For example, the *hasTable* and *hasImage* properties (see their application in Figures 4 and 5) can be used jointly with an image annotator and an information extractor from tables, in order to add new explicit data e.g. to a fishery area that is assumed to be described in a fact sheet.

A preliminary workflow to refactor the parts of fi.owl that can be interpreted as an actual fishery ontology is sketched here.

- Step 1. We need to distinguish properties referring to actual entities, vs. properties referring to descriptions of entities, vs. properties referring to information objects, vs properties referring to extrinsic, e.g. management-oriented information

- Step 2. We need to partition the class space into classes of actual entities (e.g. aquatic resources, species, fish techniques), possibly inferring also sublass axioms with the help of experts (or semi-automatically, if applicable), and classes of information objects (e.g. images, tables, lists, formats)

- Step 3. We need to redistribute the properties according to the class space partition, and to link the two types of classes appropriately. These reengineering patterns [cf. D251, forthcoming] are extracted from the InformationObjects[12] and the IRE ontologies.

- Step 4. We need to create a modular structure into the reengineered FI ontology, which possibly matches the FSDAS modular architecture, so that the alignment between parts of the FI ontology and the FSDAS ontologies is made easier.

(4) *Structural sparseness*

Issue (4) requires a finer-grained approach, which lets the ontology designer be aware of the actual patterns used for documental reasons, and to decide if we should take action to fix them or use them consistently across the ontology, or if we just need to extract those patterns that are useful to fix the issue (3). This finer-grained approach is left to future work, which will be reported in the next deliverable on month 30.

[12] http://www.loa-cnr.it/ontologies/DUL.owl

References

[AGMES] FAO. Agricultural Metadata Element Set. http://www.fao.org/aims/agmes_intro.jsp

[AIDA] http://www.fao.org/fi/figis/devcon/schema/3_6/aida.xsd

[D2.1.1] D2.1.1. Design rationale for collaborative development of networked ontologies. NeOn deliverable. February 2007.

[D2.5.1] D2.5.1. Library of formal models and design patterns for collaborative development of networked ontologies. NeOn project report. To appear.

[DC] Dublin Core Metadata Initiative. http://dublincore.org/

[DCT] Dublin Core Terms. http://dublincore.org/documents/dcmi-terms/

[EDC] Extended Dublin Core. http://dublincore.org/schemas/xmls/qdc/2003/04/02/dc.xsd

[FS] FAO. Fisheries fact sheets.
http://www.fao.org/fi/website/FIRetrieveAction.do?dom=topic&fid=16062&lang=en

[FSdic] FIGIS XML. List of elements. http://www.fao.org/fi/figis/devcon/diXionary/figisdoc3.5.html

[FSschema] XML schema for Fisheries Fact Sheets.
http://www.fao.org/fi/figis/devcon/schema/3_6/fi.xsd

[SW] SemanticWorks. http://www.altova.com/

[TBC] TopBraid Composer. http://www.topbraidcomposer.com/

Bibliography

[AGMES] FAO. Agricultural Metadata Element Set. http://www.fao.org/aims/agmes_intro.jsp

[AGROVOC] FAO. AGROVOC thesaurus. http://www.fao.org/aims/ag_intro.htm

[ASFA] FAO. ASFA thesaurus. http://www4.fao.org/asfa/asfa.htm

[BAR03] J. Barrasa and O. Corcho and A. Gomez-Perez. Fundfinder -- a case study of database-to-ontology mapping. In Proc. ISWC Semantic integration workshop. 2003.

[BAR06] J. Barrasa. Semantic upgrade and publication of legacy data, Ontologies for Software Engineering and Software Technology, 2006

[BAR07] J. Barrasa. Modelo para la definición automática de correspondencias semánticas entre ontologías y modelos relacionales. PhD thesis. Facultad de Informativa, Universidad Politecnica de Madrid. Madrid, Spain. March 2007. Isbn: 90-75176-81-3. http://eprints.eemcs.utwente.nl/7146/

[BIZ03] C. Bizer. D2R MAP - A Database to RDF Mapping Language. In Proc. of 12th International World Wide Web Conference. 2003.

[DC] Dublin Core Metadata Initiative. http://dublincore.org/

[D1.1.1] D1.1.1. Networked Ontology Model. NeOn project report. http://www.neon-project.org/web-content/index.php?option=com_weblinks&catid=17&Itemid=35

[D7.1.1] D7.1.1. Specification of user requirements on the case study. 2006. NeOn project report. http://www.neon-project.org/web-content/index.php?option=com_weblinks&catid=17&Itemid=35

[D7.2.1] D7.2.1. Inventory of fishery resources and information management systems. 2007. NeOn project report. http://www.neon-project.org/web-content/index.php?option=com_weblinks&catid=17&Itemid=35

[D7.4.1] D7.4.1. Software Architecture for managing the Fisheries Ontologies Lifecycle. NeOn project report. To appear.

[EDC] Extended Dublin Core. http://dublincore.org/schemas/xmls/qdc/2003/04/02/dc.xsd

[FA] FAO. Fisheries Fact Sheet. http://www.fao.org/fi/website/FISearch.do?dom=factsheets

[FAOdiv] CWP Handbook of Fishery Statistical Standards. Fishing Areas for Statistical Purposes. http://www.fao.org/fi/website/FIRetrieveAction.do?dom=ontology&xml=sectionH.xml

[FISTAT] FAO. Fisheries and Aquaculture Department. Statistics. http://www.fao.org/fi/website/FIRetrieveAction.do?dom=topic&fid=16062

[FS] FAO. Fisheries fact sheets. http://www.fao.org/fi/website/FIRetrieveAction.do?dom=topic&fid=16062&lang=en

[FSdic] FIGIS XML. List of elements. http://www.fao.org/fi/figis/devcon/diXionary/figisdoc3.5.html

[FSschema] XML schema for Fisheries Fact Sheets. http://www.fao.org/fi/figis/devcon/schema/3_6/fi.xsd

[GAN04WW] Gangemi A. WonderWeb Deliverable D16: “Reusing semi-structured terminologies for ontology building: A realistic case study in fishery information systems“, http://wonderweb.semanticweb.org, 2004.

[HBFSS] Coordinating Working Party on Fishery Statistics (CWP). CWP Handbook of Fishery Statistical Standards. Partially available at: http://www.fao.org/fi/website/FISearch.do?dom=ontology

[HS07] World Customs Organizations. Harmonized Commodity Description and Coding System. 2007 Edition. http://www.wcoomd.org/ie/En/Topics_Issues/HarmonizedSystem/DocumentDB/TABLE_OF_CONTENTS_2007.html

[ISO2] International Standard Organization (ISO): ISO 3166 ALPHA-2, 2006.

[ISO3] International Standard Organization (ISO): ISO 3166 ALPHA-3, 2006.

[ISSCAAP99] FAO. International Standard Statistical Classification of Aquatic Animals and Plants (ISSCAAP). Version in use until 1999 available at: ftp://ftp.fao.org/FI/DOCUMENT/cwp/handbook/annex/AnnexS1listISSCAAPold.pdf

[ISSCAAP00] FAO. International Standard Statistical Classification of Aquatic Animals and Plants (ISSCAAP). Version in use from 2000 available at: ftp://ftp.fao.org/FI/DOCUMENT/cwp/handbook/annex/AnnexS2listISSCAAP2000.pdf

[ISSCFVgrt] International Standard Statistical Classification of fishery Vessels (ISSCFV) by GRT Categories. in use until 1995. ftp://ftp.fao.org/FI/DOCUMENT/cwp/handbook/annex/annexL1ISSCFVgrt.pdf

[ISSCFV] International Standard Statistical Classification of Fishery Vessels (ISSCFV) by Vessel Types, in use until 1995. 1984. ftp://ftp.fao.org/FI/DOCUMENT/cwp/handbook/annex/annexLII.pdf

[ISSCFG] International Standard Statistical Classification of Fishing Gear (ISSCFG) ftp://ftp.fao.org/FI/DOCUMENT/cwp/handbook/annex/AnnexM1fishinggear.pdf

[ISSCFC] FAO. International Standard Statistical Classification of Fishery Commodities: Divisions and Group. ftp://ftp.fao.org/FI/DOCUMENT/cwp/handbook/annex/ANNEX_RII.pdf

[M49] United Nations. UN Code (M49). http://unstats.un.org/unsd/methods/m49/m49alpha.htm.

[ONEF] One fish topic tree. http://www.onefish.org/global/index.jsp

[ONTO101] N. F. Noy and D. L. McGuinness. Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001.

[PER05] C. Perez and S. Conrad. Relational.OWL - A Data and Schema Representation Format Based on OWL. In Proc. of APCCM 2005.

[RT] FAO. Table Selector for Reference Tables. http://www.fao.org/figis/servlet/RefServlet

[SITC3] United Nations Statistics Division. Standard International Trade Classification, Revision 3. http://unstats.un.org/unsd/cr/registry/regcst.asp?Cl=28&Lg=1

[XMLSpy] Altova. XMLSpy. http://www.altova.com/products/xmlspy/xml_editor.html